sun
microsystems

PC-NFS User's Manual

Part No. 814-1007-03 Rev.B November 1989

Copyrights and Trademarks

LEGAL NOTICE TO USERS:

Yellow Pages is a registered trademark in the United Kingdom of British Telecommunications plc., and may also be a trademark of various telephone companies around the world. Sun will be revising future versions of software and documentation to remove references to Yellow Pages.

Contents

Introduction **1**

Chapter 1 – PC-NFS Overview **3**
DOS Users and PC-NFS 4
Networks and Network Terminology 4
Servers 6
Directories and File Systems 6
Local and Remote 7
Summary 7

Chapter 2 – PC-NFS Features **9**
Transparent File Access 9
File Transfer 12
The File Transfer Program – `ftp` 12
The Remote Copy Program – `rcp` 12
Remote System Access 13
The `rsh` Command 13
The `telnet` Facility 13
File System Commands 14
Network Management 14
Network Statistics and Troubleshooting Commands 15

Chapter 3 – Files in a Networked Environment **17**
DOS Commands and Networked Disks 18
Copying Files Between Systems 18
Accessing Files from Different Operating Systems 19
  DOS Files 20
  NFS Files 20
Bridging the Systems 22
  Mapping File Names 22
  Converting Text File Formats 24
  Mapping File Attributes 25
  Running DOS Programs on an NFS Disk 27

Chapter 4 – File Locking **29**
Who Should Read This Chapter 29
PC-NFS File Locking Services 30
  Advisory File Locking 30
  File Sharing 31
  Record Locking 31
Using PC-NFS Locking Services 32
  Read-Only File Systems 33

Chapter 5 – Backing Up and Restoring Files **35**
Choosing a Backup Method 35
Backing Up Files to an NFS Server 37
Backing Up Files to Your Local Hard Disk 39
Restoring Files to Your Local Hard Disk 40
Restoring Files to an NFS Server 40

Chapter 6 – Using Printers **41**
Using Your Local Printer 41
Using Remote Printers 42
  Determining Your Printing Options 42
Redirecting DOS Printer Output 43
  If You Don't Know the Name of a Remote Printer 44
  Redirected Output 44
  Printing with `net print` 45
  Using a Sun LaserWriter 45
  Printing to Non-Sun Printers 46

Chapter 7 – Using Network Applications **47**
Introducing `telnet` 47
An Alternative to `telnet`: The `rsh` Command 48
Using `telnet` 48
  Beginning a `telnet` Session 48
  Logging in to a Host System 49
  Specifying the VT100 Terminal Type 50
  VT100 Terminal Emulation 51
  Changing Connections 52
  Ending a `telnet` Session 52
  Using the `telnet` Hot Key Feature 53
Using `telnet` Menus 55
  Initialization Menus 55
  Control Menus 57
  Help Menu 58
  File Menu 58
  Change Parameters Menu 60
  Serial Port Use 61
  The `telnet` Command Menu 61
Storing Previous `telnet` Settings 62
  When `telnet` Is Not Functioning Normally 63
Example: Mailing a DOS File 63

Chapter 8 – Commands Reference **65**
`arp` – Address Resolution Protocol 66
`chmod` – Change NFS File Protection 68
`connect` – Connect PC-NFS Through Modem 71
`dos2unix` – Force File Into UNIX Format 72
`ftp` – File Transfer Program 74
  `ftp` Commands 75
`hangup` – Disconnect PC-NFS From Modem 84
`ls` – Display File Information 85
`mv` – Rename a File 88
`myeaddr` – Display Ethernet Address 89
`net blip` – Turn Blip On or Off 90
`net join` – Join a Drive Letter to a Path 91
`net logout` – Log User Off Network 92
`net name` – Perform NFS User Authentication 93
`net pcnet` – Run PC-NFS With IBM PC-Network 94
`net pcnfsd` – Set or Display NFS Authentication Server 95

Chapter 8 – Commands Reference – ***continued***

`net print` – Print Files on Network Printers 96

`net route` – Set the Current Gateway 97

`net start rdr` & `net stop rdr` – Start or Stop PC-NFS 98

`net subnet` – Set the Subnet Mask 99

`net umask` – Set the Default NFS Permissions 100

`net use` – Mount Remote File Systems 101

`net version` – Display PC-NFS Version Number 104

`net ypdomain` – Set the Yellow Pages Domain 105

`net ypset` – Set the Yellow Pages Server 106

`netstat` – Display Network Statistics 107

`nfsping` – Determine If a System Is an NFS Server 108

`nfsstat` – Display NFS Statistics 110

`rcp` – Remote File Copy 113

File and Directory Name Formats 113

`rpcinfo` – Display RPC Information 116

`rsh` – Remote Shell 117

`showmnt` – Show Exported File Systems 119

`unix2dos` – Force File Into DOS Format 120

`ypcat` – Display Values From Yellow Pages Database 122

`ypmatch` – Find and Display Entry From Yellow Pages Database 123

Appendix A – Glossary 125

Index 133

Introduction

PC-NFS™ is a set of applications designed to help you get the most out of your PC and network environment.

This book is designed to introduce PC-NFS concepts and features, and includes a complete command reference, glossary and index. For information about installing and configuring PC-NFS as well as a troubleshooting guide and technical reference, see *Installing PC-NFS, A Guide to the User and System Administrator.*

This book, the *PC-NFS User's Manual*, introduces networks and terms in Chapter 1, and provides an overview of PC-NFS features in Chapter 2. Chapter 3 discusses files in a networked environment and Chapter 4 describes file locking under PC-NFS. Chapter 5 describes backing up and restoring your files across the network. Chapter 6 tells you about printing files from PC-NFS. Chapter 7 describes the network applications, `telnet` and `rsh`. Chapter 8 is the Commands Reference; it lists all PC-NFS commands in alphabetical order.

Also available are PC-NFS LifeLine, electronic mail and backup for use with PC-NFS, and The PC-NFS Programmer's Toolkit. For more information about these products, contact your Sun representative.

This manual uses:

This typeface for emphasis, to introduce a new term, for manual titles, or to indicate a word that you must supply a value for. For example: A *gateway* is a computer that links networks together.

`This typeface` to indicate command or program names, or output on a screen. For example: The `hangup` command disconnects you from the *gateway*.

`This typeface` to indicate what you should type. For example: Type **`connect`** at the C> prompt.

PC-NFS Overview

1

This chapter introduces the personal computer version of Sun Microsystems's Network File System (NFS™). PC-NFS is a PC networking product that allows you to:

- Share your files with other DOS users on your network, without exchanging diskettes.
- Transfer files between systems using commands such as the DOS `COPY` command, or the PC-NFS commands `rcp` or `ftp`.
- Share files with users of different operating systems in your NFS network, including UNIX® operating systems and VMS®.
- Use the file sharing and locking services provided by DOS 3.1 (but not the NETBIOS services provided by networks such as Microsoft MS-NET® or IBM PC Network™).
- Print files on the network's printers, including those supporting PostScript®.
- Remotely log into non-NFS systems using `ftp`, `rsh`, and `telnet`.
- Use selected networking commands compatible with the Berkeley 4.2 BSD UNIX operating system.
- Access all of these facilities either directly on an Ethernet® network or over a serial line, such as a phone line or RS232 line.

If you are familiar with networks and their terminology, you may want to skip the rest of this chapter and proceed to Chapter 2, *PC-NFS Features.* The remainder of this chapter discusses PC-NFS from a DOS user's point of view and introduces the concept of a network and its associated terminology.

DOS Users and PC-NFS

As a DOS user, the usual way to transfer a file to another DOS user on another computer system is to copy the file to a diskette and hand-carry the diskette to the other system. If you could link the two systems you could not only make the file transfer easier, but could also do the following:

- Read directories and files that reside on another non-DOS computer
- Run applications on your computer that reside on another computer
- Use a laser printer connected to another computer
- Back up files and directories to shared resources, such as magnetic tape drives
- Access databases that are too large for one PC-based system

PC-NFS is a set of applications and commands that gives you the ability to do all of these tasks. Moreover, it gives the ability to connect to machines that do not use DOS.

To understand how PC-NFS accomplishes these tasks, you must be familiar with networks. The next section provides a brief introduction to networks.

Networks and Network Terminology

A *network* is a mechanism that links two or more computers to share resources. The most commonly shared resources are:

- Information, in the form of files
- Printers, such as a laser printer
- Processing power, such as running a program on another machine

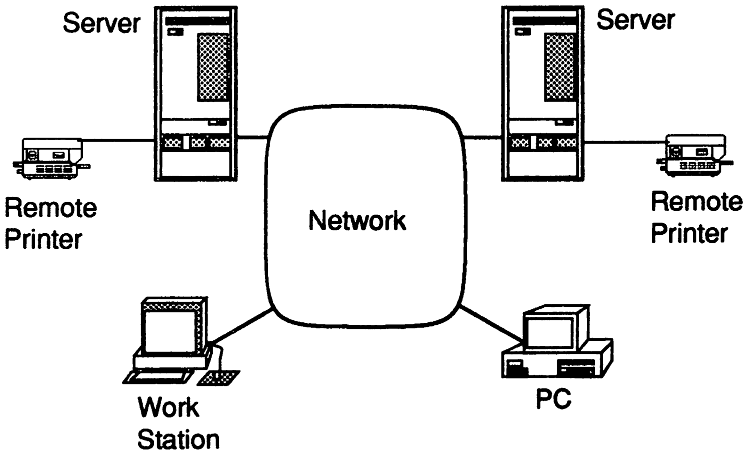
A network links the various hardware devices (computers, printers, and backup devices) you want to share. It requires both software and a physical connection between the linked devices. The physical link is a line, not unlike a telephone line, and the software is some type of network software package.

There are different types of computer networks differentiated at the basic level by the type of wire and electronics used to connect the computers. At this level, PC-NFS works with Ethernet networks. Instead of Ethernet, you can use the serial port on your PC to connect your PC to a machine on a network that supports NFS. The network can be just your machine and the machine running NFS.

The conventions governing how networks communicate in general are *protocols*. PC-NFS uses a particular set of protocols, called the Internet Protocol (IP).

A network looks something like Figure 1-1.

Figure 1-1 *Networked Computers*



Each hardware device on the network must have a unique name so that the software can recognize it. This unique name is the device's *host name*. Since you may use any computer on the network, you must have a unique name also. Your unique name is your *user name*.

When you want to use a computer on a network, you must identify yourself to the computer. This process of identification involves entering your user name and usually a password. This identification procedure is a *log in*. Once logged in, you can use the various resources available to you on the network. Usually a person, the *system administrator*, is designated to keep track of network usage and resolve any problems. This person can give you the details of the log in procedure for your network.

You system administrator can tell you if your network contains a gateway, ARP, or Yellow Pages. A *gateway* is a physical connector between two or more networks. ARP (Address Resolution Protocol) is a network protocol that helps keep Ethernet and Internet addresses straight. The Yellow Pages is a network service that provides information about Ethernet and Internet addresses, names used for mail, host names, groups, and passwords.

Among the resources you share are programs or sets of programs that perform tasks such as word processing. Each of these programs is an *application.*

Printers, backup devices, and applications often reside on a remote system for sharing purposes. The next section discusses how these devices operate in the network environment.

Servers

In a network, any computer that performs a service for another computer is a *server* system. Computers that use of any of these server systems are that system's *clients.* A server system usually takes the name of the application or task it is performing. Systems set aside as storage for large numbers of files are *file servers.* Other systems serve as control machines to shared printers and are *print servers.* In some cases a machine performs more than one server function, such as being both a file server and a print server.

Within PC-NFS, a PC running DOS can never be a server. It is always a client, which must access at least one NFS server. Consult your system administrator for the availability of servers on your network.

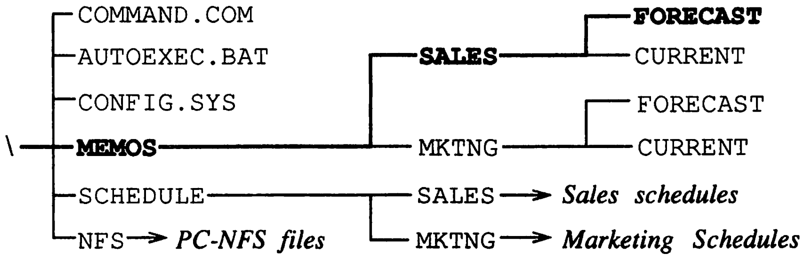
Every file server contains a large number of files that must be organized in some fashion to provide quick and easy access. The next section discusses the mechanics of file organization in general.

Directories and File Systems

Files are the basic units of information storage for a computer system. Most computer systems organize files in directories. In a DOS environment, this organization is a *directory tree structure.* In a UNIX environment, the organization is called a *file system.* For both environments a directory can contain files and other directories in a hierarchical manner

You specify the location of any particular directory, or file within a directory tree, by using a *path name.* The path name begins with the top-level directory name and includes all directories under that directory leading to the desired directory or file, as shown in Figure 1-2.

Figure 1-2 *Hierarchical Directory*



Path: **\MEMOS\SALES\FORECAST**

The term *file system,* as used here, should not be confused with the more general usage of the term which includes the directory structure and the software that manipulates files within that structure. Throughout this book, the term file system always takes the narrow meaning of a directory structure. When the term applies more generally, this book uses the term *file management system.*

Networks may combine several systems that use different terms for the organization of directories. However, PC-NFS uses the UNIX term file system. A network gives you access to your own file system and those on other machines. The next section introduces the terminology for different systems in a network.

Local and Remote

The term *local* refers to any file that is on your system or any device that is physically attached to your system. Devices and files existing on machines on the network, other than your own computer, are *remote.* So a printer attached to a server on the network is a *remote printer.* A file system on a network server is a *remote file system.*

There are several ways to get information from a remote system. The most direct method is to log in to the remote system from your local machine. This procedure is a *remote log in.* Once you log in, you use your computer as a terminal on the remote machine.

Summary

This chapter covered the basic idea of a network. It also gave some glimpses of how PC-NFS fits in the network environment. The next chapter describes the various features of PC-NFS.

PC-NFS Features

2

Personal computer users can use PC-NFS to access a network while running applications under DOS. It creates an environment in which file systems on remote computers are treated as separate local disk drives and remote printers are treated as local printers. Since the remote computers and printers are shared resources, they can be accessed by more than one user on the network at a time. In a network you need to keep users, who may be sharing the same resource, from bumping into each other. PC-NFS provides this service and many others.

There are six basic services PC-NFS performs:

- Transparent File Access
- File Transfers
- Remote System Access
- File System Commands
- Network Management
- Network Statistics and Troubleshooting Commands

This chapter discusses each of these capabilities and details the commands available in each. For details on the usage of any of the commands, see Chapter 8, *Commands Reference*.

Transparent File Access

Using PC-NFS, remote computers and printers are available to you as though they were local directories and printers. You use all the familiar DOS commands to interact with them.

In order for you to access remote printers and file systems on your network, you must first link your system with the remote device. You can use either the configuration program or the `net use` command to create the needed link. If you use the configuration program to establish the link, it re-establishes the link every time you reboot. You can create a temporary link with the `net use` command (see Chapter 8). To link a file system on a remote computer to your local system, you *mount* the file system. You also mount remote printers using the configuration program or `net use`.

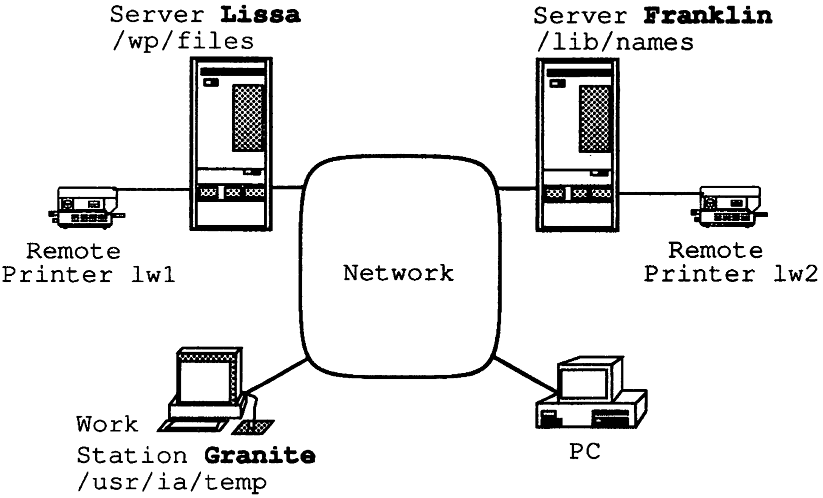
Once you mount the file systems and devices, you access them as though they are part of your own system. The following example illustrates how your personal computer and remote devices interact using PC-NFS.

Suppose you want to access three file systems on three different remote computers, as well as two remote printers. The remote computer systems names and the corresponding path names for the file systems are:

Computer Name	**Path Name for File System**
Lissa	`/wp/files`
Granite	`/usr/ia/temp`
Franklin	`/lib/names`

Your network looks something like Figure 2-1.

Figure 2-1 *Example Network*

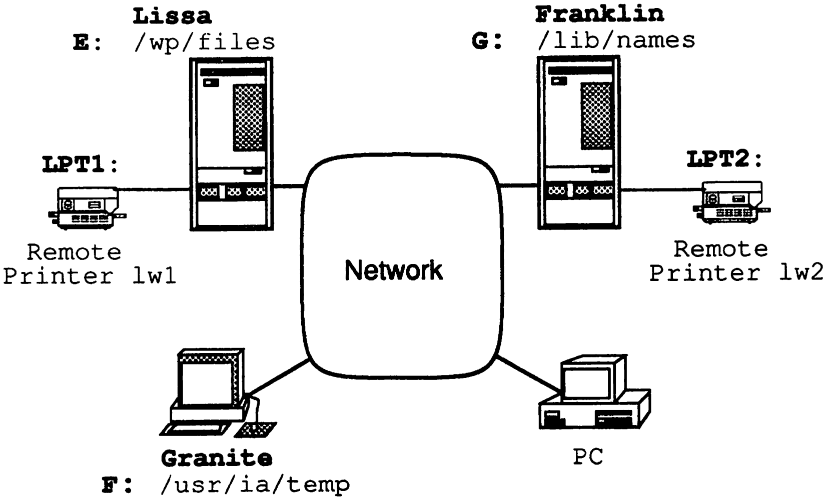


PC-NFS interacts with each remote file system as though it were a separate disk drive on your local system. The DOS operating system recognizes three parallel printing devices, `LPT1`, `LPT2`, and `LPT3`. You can associate a remote printer with any of these devices.

In this example, you configure the remote file systems and printers as in the following table:

Computer	Name	As seen by DOS on PC
Lissa	`/wp/files`	Disk drive E
Granite	`/usr/ia/temp`	Disk drive F
Franklin	`/lib/names`	Disk drive G
Remote Printer 1	`lw1`	LPT1
Remote Printer 2	`lw2`	LPT2

Figure 2-2 *Example Network with DOS's Names*



You treat these disk drives and printers as though they were separate local devices operating under DOS (See Figure 2-2.) So, if you want to list the contents of file system `/usr/ia/temp` on **Granite**, you enter:

```
C>DIR F:
```

and even though **Granite** may be a UNIX machine, you see the directory listing in the familiar DOS format. Your applications that have printer interfaces built into them send output to the remote printer lw1 if you configure the application's printer as `LPT1`.

File Transfer

When you move a file from one computer to another electronically you accomplish a *file transfer*. Within PC-NFS there are two facilities for file transfer:

- ftp
- rcp

Each of these facilities transfers files, although each accomplishes the transfer in its own way. You can also use the DOS COPY command. The next two sections briefly describe the functionality of ftp and rcp.

The File Transfer Program - ftp

The File Transfer Program (ftp) is an industry-standard application that provides a wide range of options for transferring files. Among the options are:

- Accessing the file system of the remote system
- Sending files to a remote system
- Getting files from a remote system
- Transferring files using any of a variety of file formats including ASCII, binary, DOS, or UNIX format
- Performing actions on multiple files and directories, including the use of wildcards

Specific software must be running on the server before you can use ftp.

The Remote Copy Program - rcp

The Remote Copy Program (rcp) copies files from one host to another. It can access directories and subdirectories implicitly but has no ability to change permissions, file names, or other file attributes. It requires the UNIX operating system on the remote device.

The rshd software must be running on the server before you can use rcp.

Remote System Access

When your machine is part of a network of machines, you may want to get information from the other machines, or even at times log into another machine to perform a task. PC-NFS offers two very different means of remote access:

- `rsh`
- `telnet`

Each of these facilities provide remote access, although each accomplishes the access in its own way. The next two sections briefly describe these applications.

The rsh Command

The `rsh` command offers a direct means of executing a command on a remote system. You identify the remote system and the command you wish to execute. Do not use interactive commands, that is, commands that prompt you for more input, since interactive commands do not work as you would expect with `rsh`.

To use `rsh` requires specific software, `rshd`, on the server.

The telnet Facility

The `telnet` facility is an industry standard application that uses a VT100 emulation to connect to a remote computer over the network. Using `telnet` you can:

- Log in to remote systems
- Switch between your remote session and DOS
- Choose network or com port interfaces
- Access machines not running NFS, if they have `telnetd` installed

To use `telnet` requires specific software on the server.

File System Commands

These are commands that manipulate file attributes and information. The commands are:

dos2unix – changes the format of DOS text files to the UNIX text file format.

unix2dos – changes the format of UNIX text files to the DOS text file format.

ls – lists the contents of a specified local or remote directory.

mv - changes the name of a specified file to a new specified name or moves the file to a new directory.

chmod – permits you to change who has access to files and directories and the type of access they have. The command also controls whether or not a file is executable, in the UNIX sense of the word.

Network Management

There are sixteen commands that display, set, and control various aspects of the network environment. A short description of each command follows. For more information, see Chapter *Commands Reference* in this book.

net blip – controls the network *activity indicator.* The activity indicator ("blip") is the flashing square in the upper right-hand corner of the screen that signals network activity. It is similar to the disk drive activity light found on many personal computers.

net join – displays and controls joining of network devices.

net logout – logs the user off the network.

net name – displays the name and Internet address of a PC, Yellow Pages server, authentication server, and gateway system. It also allows you to log in.

net pcnet – controls whether PC-NFS runs concurrently with the IBM PC Network.

net pcnfsd – displays and sets the name of the authentication server.

net print – prints files on network printers.

net route – displays and sets the name of the current gateway.

net start rdr – starts PC-NFS.

net stop rdr – stops PC-NFS.

net subnet – sets the subnet mask.

net umask – sets default file permissions. See Chapter 3, *Files in a Networked Environment* for information on file permissions.

net use— displays or mounts a remote file system.

net version – displays the current version number of PC-NFS.

net ypdomain – sets the name of the Yellow Pages domain.

net ypset – sets the authentication server and the Yellow Pages server to the system you specify.

Network Statistics and Troubleshooting Commands

The commands described in this section provide information about either the network, NFS, Yellow Pages, or remote machines. For more information about these commands, see Chapter *Commands Reference* in this book.

arp – displays Internet and Ethernet address pairs that are stored inside PC-NFS.

netstat – displays network interface or protocol statistics.

nfsping – determines if a remote machine is an NFS server and if the machine is available on the network for communications.

nfsstat – displays NFS internal protocol statistics.

rpcinfo – displays information concerning which remote procedure services are available on remote machines.

showmnt – displays information concerning which files systems a server has exported.

ypcat – displays values in a Yellow Pages database.

ypmatch – displays a selected entry in a Yellow Pages database.

Files in a Networked Environment

3

One of the key features PC-NFS provides is its ability to operate in a heterogeneous environment of operating systems and machines, and, to a great extent, provide *transparent file access*. Transparent file access means that you can access remote files as if they were local to your system.

Using PC-NFS, you can work within the DOS operating system and share work with users of other operating systems, such as the UNIX operating system and VMS. You can greatly expand your disk storage by using remote drives on the network. If you are comfortable working in different systems, you can work in whichever environment offers the best tools for the task at hand, while you leave your files in one place.

PC-NFS itself is fairly invisible to you. When you boot your DOS system with PC-NFS installed, you remain in your DOS environment. Your DOS commands work, your DOS applications work, and you can continue as if PC-NFS were not there. What PC-NFS provides you, however, is a way to greatly expand your environment by allowing you to access remote resources as if they were local to your system. You can create and store files on network file systems and access them as easily as you now access your own DOS files.

DOS Commands and Networked Disks

You can use DOS commands and applications to work with your remote files as well as your local files. However, DOS commands that affect disks do not work on network disks. Remember that, although you have access to network resources, these resources belong to other systems with different operating systems. You can manipulate your files on these remote disks, but you cannot manipulate the disks themselves. Therefore, the following DOS disk commands do not work with mounted file systems:

`CHKDSK`	Analyzes the directory use and storage capacity of a disk
`DISKCOMP`	Compares one disk to another
`DISKCOPY`	Copies one disk to another
`FDISK`	Creates, changes, deletes, or displays partitions on a hard disk
`FORMAT`	Prepares a disk for use
`JOIN`	Connects a disk drive to a directory on a different drive. This is similar to the UNIX `mount` command. Use `net join`.
`LABEL`	Creates, changes, or deletes a volume label on a disk
`RECOVER`	Recovers files from a disk that has defective sectors
`SUBST`	DOS 3.1 SUBST causes a drive specification to refer to another drive. `SUBST` does not work in DOS 3.2.
`SYS`	Transfers operating system files from one disk drive to another

These commands continue to work with your local hard disk and diskettes.

Copying Files Between Systems

With your expanded environment, you may want to copy your files to a mounted file system to save space on your local disk or to make them available to other network users. You may also want to make a local copy of files residing on a mounted file system. PC-NFS provides you with many different ways to move files between systems. The three most common are:

- DOS `COPY`
- `rcp` -the Remote Copy Program
- `ftp` - the File Transfer Program

You are probably already familiar with the DOS COPY command. If you're not, refer to your DOS reference manual. PC-NFS enables you to use the DOS COPY command to copy files from one directory to another, and the directories may be on local or remote disks. The DOS COPY command is probably adequate for most of your file transfer needs.

PC-NFS provides `rcp` and `ftp` to allow you to exchange files with systems that do not yet support NFS.

The `rcp` command allows you to transfer files to and from a UNIX system. One advantage of the `rcp` command is that it allows you to copy files to or from a remote site regardless of whether you have explicitly mounted the remote file system. Another advantage of `rcp` is that it allows you to copy nested directories with a single command. For details about using `rcp`, see Chapter *Commands Reference* in this book.

The `ftp` program is a generic file transfer utility that allows you to transfer a file to any system on your network that provides TCP services, but not necessarily NFS. (TCP stands for Transmission Control Protocol, and is a protocol implemented for many networked systems.) However, `ftp` requires you to begin a *session* with the remote system. An `ftp` session allows you to establish a connection with a remote system and perform many remote file operations. `You can` copy more than one file at a time, but you cannot copy directories. See Chapter 8 for details on using `ftp`.

Accessing Files from Different Operating Systems

Every operating system available to you in your network environment has files and a file system. The actual way in which data is represented in a file is called the *file format*. The structures that the file system uses to manage files are referred to as *file attributes*. File formats and file attributes vary from operating system to operating system.

When you use one operating system to access a file created by a different operating system, you need to be aware that the different file format and file attributes must be translated, or *mapped*, into a file format and file attributes understandable by the operating system you are using.

The DOS COPY command, and the PC-NFS commands, `rcp` and `ftp`, copy files from your DOS system to a remote network site and bring remote files to your DOS environment. Each one of these facilities preserves the file format of the file it copies. Therefore, when you use DOS COPY to copy a DOS file to an NFS disk, the copy you create on the NFS disk is in DOS format. If you use DOS COPY to copy an NFS file onto your local disk, the file you create has the characteristics of an NFS file.

DOS Files

In order to understand file formats and file attributes, and their importance to operating systems, start by looking at the characteristics of DOS files:

- DOS files have file names that are up to eight characters long and have an optional extension that is three characters long. DOS does not allow the following characters in a file name:
  `" . / \ [ ] : | < > + = ; ,`
- DOS file names are *case-insensitive*: all lower-case letters are silently translated into their upper-case equivalent. This means that the file names `abc`, `Abc`, and `ABC` are equivalent within DOS. In effect, all `DOS` file names are uppercase.
- DOS directory names follow the same rules as DOS file names. DOS path names consist of directory names separated by backslashes (\) followed by the file name (`directoryname\directoryname\filename`).
- DOS uses a two character sequence, carriage return line feed, to terminate a line in DOS files. DOS inserts the CONTROL-Z character at the end of DOS files to indicate end-of-file. This is the DOS file format.
- DOS uses a file attribute byte to track information about the file. This byte includes information such as whether the file is *read-only* and whether the file is a *hidden file*. For a full discussion of DOS attributes, consult your *DOS Technical Reference.*

DOS is an operating system designed with the individual user in mind. DOS cannot distinguish one user from another. The kind of information that DOS needs to manage files is only the information needed in a single-user environment.

Other operating systems, such as the UNIX operating system and VMS, were designed as multiuser environments. Their file systems were designed to include information about who is allowed to use a file as well as what kind of access a user has to a file. NFS file support is modeled to a large extent on the UNIX file system.

NFS Files

When you compare NFS files to DOS files, you see that, although both NFS and DOS file systems have similar components and similar functions, NFS has added structures that facilitate protecting your files in a multiuser environment.

Compare these NFS file characteristics with the characteristics of DOS files discussed in the previous section:

- NFS allows file names that consist of any printing characters except the slash (/) and may be up to 255 characters long.
- NFS supports file names that are *case-sensitive*. This means that NFS sees the file names `abc`, `Abc`, and `ABC` as distinct names.
- NFS directory names follow the same rules as file names. NFS path names consist of directories separated by slashes (/) followed by a file name (`directoryname/directoryname/filename`).
- NFS supports two types of file attributes to protect files: permissions and user classes. *Permissions* refer to the kind of access users have to a file (read, write, or execute). *User classes* refer to the kinds of users who have access to the file. Together permissions and user classes determine who has what kind of access to a file. The next section more fully explains permissions and user classes.

NFS File Attributes. For every file and directory, NFS supports three different types of permissions and three different types of user classes. NFS permissions and user classes are modeled after UNIX permissions and user classes.

Permissions, also called *modes*, include the three categories:

read Allows a user to inspect a file.

write Allows a user to alter or erase a file.

execute Allows a user to execute a file as a system command.

User classes include:

user The person who created the file or directory and who can assign permissions to it.

group A collection of users associated by a *group name*.

others All NFS system users other than yourself (the user) and users in your group.

For each of the three user classes, *user, group*, and *others*, you can set the three available permissions: *read, write*, and *execute*. Thus there are nine possible permissions that you can set for each file. Depending on your needs, you can allow yourself, your group, and all other users to inspect or alter your files and directories, or you can prohibit such access.

Bridging the Systems

Accessing files created on one operating system from another operating system can be difficult. There can be problems with differences in the file format, the file name conventions, and the file attributes. Fortunately, PC-NFS does most of the work for you, and provides you with tools to help you do the rest. The following sections describe how you can access UNIX files from your DOS system, and what you need to do to make your DOS files accessible from a UNIX system.

Mapping File Names

DOS file names are legal UNIX file names. Therefore, when you try to access your DOS file from a UNIX system, the UNIX system understands the DOS file name. However, UNIX file names can be longer than DOS allows, and they may contain characters illegal in DOS file names. PC-NFS uses a set of rules to translate UNIX file names into legal DOS file names. It keeps a record of the translation, a *map*, in a table. The table contains only the 64 most recent mappings. This translation process is spelled out in the next several sections.

File Name Mapping Summary. Since UNIX file names consist of predominantly lower-case letters, the conversion process inverts the case of all letters before attempting to determine whether the name is a legal DOS file name. Thus the NFS file name `phase1.c` becomes the legal DOS name `PHASE1.C`.

If the conversion to uppercase letters makes a legal DOS file name, eight characters with an optional three character extension separated by a period, the conversion is done. If not, the following further steps are necessary.

The PC-NFS file mapping uses the first five characters of the original file name in upper case, followed by a tilde in the sixth character position. The seventh and eight characters are random characters generated by the mapping to insure that each name is unique. If any of the first five characters are not legal characters, or do not exist, they are replaced by tildes. Finally, any legal DOS extension is appended.

Note that this algorithm always yields an eight-character name with a tilde replacing any character in position six. You can list all mapped names using DOS wildcard characters and the `DIR` command like this:

```
G>DIR ?????~??.???
```

Mapping DOS File Name Extensions. If a file name extension is legal in DOS after lowercase to uppercase inversion, PC-NFS preserves the extension. If inversion yields an illegal DOS extension, PC-NFS includes the extension in the mapping algorithm described above. The preservation of legal extensions, such as `.OBJ`, allows programs such as compilers to synthesize file names by varying the extension, even when the file name is mapped.

When PC-NFS maps an extension from the DOS form back to the UNIX form, PC-NFS restores the original file name and extension.

Examples of Name Mapping. The following table shows examples of name mapping. In Table 3-1, `XX` represents the two-character *cached name handle* used by PC-NFS:

Table 3-1 *Name Mapping Examples*

Sample UNIX file name	Mapped DOS file name	Notes
`abc123.def`	`ABC123.DEF`	No mapping required.
`a`	`A`	No mapping required.
`A`	`A~~~~~XX`	Uppercase mapping.
`a_long_name`	`A_LON~XX`	Mapping usable characters.
`AB.c`	`AB~~~~XX.C`	Only two usable characters.
`Ab.c`	`AB~~~~XX.C`	Note that the `XX` value will be different from the last example, since this is a distinct mapping.
`a.b.c`	`A~B~~~XX.C`	Illegal because of syntax; the first embedded period is replaced by a tilde; the extension is OK.
`abcd.efgh`	`ABCD~~XX`	The extension is illegal.
`.login`	`~LOGI~XX`	A leading period always makes a name illegal.

Using Mapped Names – The mapped file names that PC-NFS creates are *temporary* handles used *only* by PC-NFS. Name mapping does not produce the same result each time, due to the randomness of assigning characters seven and eight. Therefore, you should not build mapped names into your applications. You can use the PC-NFS command `ls` to show you what mapping is in effect at any given time. See the sample `ls` output on page 26.

If you want to refer to the file name from a program, rename the file to a legal DOS file name. Use the DOS `RENAME` command, or the PC-NFS `mv` command to rename the file. For details about the `mv` command, see Chapter 8.

Converting Text File Formats

DOS text file format differs from UNIX text file format. UNIX lines are terminated by the line-feed character; DOS lines are terminated by the two-character sequence, carriage-return line-feed. Because of this difference in file formats, files created by a UNIX system are missing necessary carriage returns when accessed by DOS, and files created by DOS have superfluous carriage returns when accessed by a UNIX system.

As an example, consider the following UNIX text file called `NEWS`:

```
%cat NEWS
The most fundamental of the Sun Network services
is the Network Filing System (NFS), an
operating-system independent service which allows
users to mount directories across the network
and then to treat those directories as if they
were local.
```

If accessed by way of a DOS application program such as TYPE, this text appears strange:

```
C>TYPE NEWS
The most fundamental of the Sun Network services
                                                is
the Network Filing System (NFS), an
                                   operating-syste
m independent service which allows
                                  users to mount d
irectories across the network
                             and then to treat tho
se directories as if they
                         were local.
```

Likewise, native DOS text appears strange when you display it after you copy it the UNIX disk.

PC-NFS provides programs to convert these files. The programs are dos2unix and unix2dos, which are documented in Chapter 8.

Mapping File Attributes

DOS and UNIX file attributes differ as well. Mapping file attributes is difficult because there is not a one-to-one correspondence between DOS file attributes and UNIX file attributes.

While the UNIX operating system has explicit read, write, and execute permissions for every file, DOS permits you to read and write most files by default. You can specify read only access in DOS using the DOS ATTRIB command. If you have read access to a DOS file, you can also execute that file. Under the UNIX operating system, you need explicit execute permission to execute a file.

DOS has no sense of file ownership. However, in a networked environment, you need the security associated with being the owner of your file. When you access a DOS file from a UNIX system, PC-NFS creates permissions and user classes for your DOS file and makes you the owner. Likewise, when you access a UNIX file from DOS, PC-NFS maps all of the file's permission and user class information into a single DOS file attribute. For more information about how PC-NFS supports DOS file attributes, see the Technical Reference chapter in *Installing PC-NFS, A Guide to the User and System Administrator.*

Various applications packages and commands map file attributes in different ways. If you use DOS `COPY` to move a read-only file from your DOS system to a UNIX system, your UNIX file will have read-write-execute permission. If you use `rcp` to move the same DOS file, your UNIX file will be read-only.

How PC-NFS Uses File Attributes. Every user on a UNIX server system is assigned a *user name* and (optionally) a *group name*. The user name is what the server system uses to identify you and to assign ownership to the files you create.

PC-NFS allows you to specify a user name that identifies you in all remote file operations. If you do not specify a user name, the default name `nobody` is used. Normally the program assigns your user name using the `net name` command.

PC-NFS supports membership in only one group. When you log in using `net name`, PC-NFS retrieves your group name. If you need to change your group membership, your system administrator must make that change for you. When this has been done, you can rerun `net name` and your new group name will be assigned.

Files and directories you create have the user name and group name assigned to you and you own the files. Each time you create a file or a directory, PC-NFS assigns permissions to that file or directory. These are known as *default* permissions.

Changing NFS File Attributes. You can display information about your files, including their permissions, using the PC-NFS `ls` command. The following is sample output from `ls`:

```
E>LS
A                 15   8-12-87    3:51p  U:rw-
A~~~~~CJ          20   8-12-87    3:51p  U:rw-  A
LIST              96   8-19-87    7:43a  U:rwx
12345678 123       8   8-14-87    1:59p  U:rw-
12345~DB          15   8-14-87    2:00p  U:rw-  1234567894
DUMMY             15   6-05-87    2:06p  U:rwx
RCPNE~6A          15   8-06-87    9:27a  U:r--  rcpnewdumy
READDY            15   8-05-87   11:03a  U:r--
READY~4A          15   8-05-87   11:26a  U:r--  readyornot
```

The `ls` command lists the mapped file name, the size of the file, the date the file was created, the time the file was created, the permissions, and the original NFS file name. If the original NFS file name was all lower-case letters and otherwise, a legal DOS file, `ls` does not list it. See Chapter 8 for details on how to use the `ls` command.

To change the permission of a file or directory that has already been created, use the PC-NFS `chmod` command. For details about using `chmod`, see Chapter 8.

You can change default permissions, those assigned to your files and directories automatically, with the PC-NFS `net umask` command, as described in Chapter 8.

Permissions for NFS Directories. Assigning read and write permission to files in your NFS server's directories is not enough to protect or to use the files. NFS *directories* must have execute permission for their contents to be accessible. If you do not have execute permission for a directory, you cannot copy files into the directory, make the directory the current working directory, or rename, inspect and modify files in that directory.

On the other hand, if a directory is executable and writable, you can rename or delete files in that directory, even if you do not have write permission for the file. The result is that an unauthorized user can use a DOS utility such as `EDLIN` to change the contents of your directory. When you use `EDLIN` to edit a file, it first renames the file (to *file*`.bak`) and then creates a new file with the original name containing the changes. Because of this behavior, *you should make directories (as well as the files contained in them) write protected if you wish to protect files from unexpected modification.*

DOS pipes may behave differently on a local disk than on a remote disk. Pipes are implemented as files in the root directory. In order for them to work properly, the user must have read and write access to the root directory of the remotely mounted file system.

Running DOS Programs on an NFS Disk

You can store DOS programs on the NFS server disk along with DOS text files, server programs, and server text files. DOS, however, does not check the permission of a file to determine whether it can execute the file. DOS identifies executable files by their file name extensions. Any DOS file with a name ending in `.COM`, `.EXE`, or `.BAT` is a DOS executable file. With PC-NFS, DOS programs with appropriate file name extensions can be stored on the NFS disk and executed by DOS in the usual DOS way.

File Locking

4

This chapter describes PC-NFS *File Locking Services*. PC-NFS File Locking Services are supported only on networks with a server running the SunOS system, release 3.0 or later.

PC-NFS File Locking Services implement the DOS 3.1 file sharing and locking functionality through the NFS network locking services supported by the SunOS system. This means that PC-NFS File Locking Services are DOS-compatible and that you can successfully use them with PC applications written prior to DOS 3.1. File Locking Services (and other optional services) will be supported by other NFS server vendors in the future. If this is important to you, contact the appropriate vendors for details.

Who Should Read This Chapter

You should read this chapter if you:

- Work with shared data bases, program source files, word processing documents, or spreadsheets
- Use PC applications that rely upon the file sharing and locking services provided by DOS 3.1 (but *not* by NETBIOS or MS-Net)
- Want to provide maximum DOS compatibility for future PC applications

You don't need to read this chapter if you:

- Use non-Sun server systems
- Run a SunOS release prior to 3.0
- Expect to be the only user of PC-NFS on your network
- Do not share data files with other PC-NFS users
- Use PC-NFS for `telnet` or print redirection only

If you still aren't sure whether this chapter applies to you, read the following section for an overview of file-locking concepts.

PC-NFS File Locking Services

As long as you remain in your stand-alone DOS environment, you don't have to worry about controlling access to your files. However, as part of a network, your data becomes vulnerable to other users, and you may potentially damage other users' data.

File protection mechanisms such as permissions and user classes offer protection from certain users. But what kind of protection do you have when someone who has legitimate access to your file tries to update that file at the same time you do?

To protect files from this kind of inadvertent yet disastrous interaction, PC-NFS supports *file locking*. File locking includes *file sharing* and *record locking* as defined by IBM and Microsoft and released as part of 3.1 MS-DOS and PC-DOS.

Advisory File Locking

PC-NFS File Locking Services support *advisory file locking*. Advisory file locking causes the system to verify only that lock requests are granted in a consistent manner. PC-NFS does not support *mandatory file locking*. Mandatory file locking causes the system to verify *every* access to a file against existing locking permissions.

Advisory file locking is consistent with the recommendations of the *DOS Technical Reference*, but does not implement the full functionality of DOS 3.1.

Advisory file locking allows you to maintain a consistent database for your distributed applications. However, this type of file sharing is enforced correctly only if *all* PC-NFS client systems accessing a given NFS file system *explicitly request* File Locking Services with the `net use` command or the `nfsconf` program.

Advisory file locking means that the network *does* detect inconsistent lock requests. However, advisory file locking *does not prevent* other users from writing to your file. Users still have to check the file's file locking permissions. To guard against these access conflicts, use the DOS file sharing modes, as described in the *DOS Technical Reference*, when you open and create files.

File Sharing

File sharing allows you to open a file and specify both *how you intend to access the file* (read-only, write-only, or read-and-write), and *what type of access is denied to other users* while you have the file open.

For example, you can code a request that says "I'd like to open and read file `MYDATA.XYZ`. While I have it open, nobody else should be able to write to it." In DOS 3.1 terminology, this request is for *Read Access* in *Deny Write Mode*. If the system cannot grant this request (for example, if someone else has already opened the file for writing), your request to open the file is denied.

By default, DOS opens files in *compatibility mode*. Compatibility mode provides *Deny Write* access if you want to read the file, and *Deny All* (exclusive) access otherwise. Compatibility mode thus ensures full support for your pre-DOS 3.1 applications.

Note that PC-NFS supports all of the DOS file sharing modes. For more information about each of these modes, see the *DOS Technical Reference*.

Record Locking

File sharing provides a reasonable degree of security at the file level. However, many applications require that a common database be updated by multiple clients (for example, a transaction-based stock control application). For this kind of situation, DOS 3.1 supports *record locking*, also known as *byte-range locking*. Record locking lets you temporarily deny other users access to a particular *section* of a file.

To update a particular data base entry in a PC-NFS environment, you can code a sequence of DOS requests to:

1. Lock the area of the file that spans the entry (this might require multiple locks if the entry is fragmented)
2. Read the entry into memory
3. Update the in-memory copy
4. Rewrite the data to the file
5. Release the lock(s)

If the system cannot grant a lock after a (programmable) number of attempts, the lock request fails.

The *DOS Technical Reference* cautions you not rely on read or write access being denied for a locked range. Instead, you should use explicit locks around any critical byte ranges in a file.

Since PC-NFS supports advisory locking rather than mandatory locking, attempts to read and write locked ranges *succeed* although attempts to lock these already-locked ranges fail. Therefore, when you are working with PC-NFS File Locking Services, you need to ensure that all programs accessing a common file abide by locking protocols.

Using PC-NFS Locking Services

In order to use PC-NFS File Locking Services, the locking services software must be installed on your server. If you encounter any difficulty using the file locking services, consult the "Troubleshooting PC-NFS" chapter of *Installing PC-NFS, A Guide to the User and System Administrator.*

You can invoke PC-NFS File Locking Services in two modes: *share* and *mustshare*. Specify share mode if you prefer to use File Locking Services, but want to proceed even if File Locking Services are not available. You can use share mode if you expect file access conflict to be unlikely.

To guard against *any* risk of file access conflict (to maintain database consistency, for example), you specify *mustshare* mode. Using *mustshare* prevents all access to the drive if the File Locking Services detect a Lock Manager failure.

You can use either the `net use` command or the `nfsconf` program to specify locking services for mounted file systems. If you want to use `nfsconf` see Chapter 2 of the *Installing PC-NFS* manual. Following are directions on how to use the `net use` command.

To invoke File Locking Services, you use a *sharing switch* with the PC-NFS `net use` command or from within `nfsconf`, when you mount an NFS drive. Use the `/SHARE` or `/SH` switch to specify that, if possible, you want to use PC-NFS File Locking Services with the NFS file system you specify. When you specify the `/SHARE` or `/SH` switch, the system mounts the NFS file system in *sharing mode*. Sharing mode means that:

- Lock requests are passed to the Lock Manager on the server for verification.
- File `OPEN` and `CREATE` requests are subject to file sharing, as described earlier in this chapter.

Note that all DOS 3.1 sharing modes are supported over the network. The default, without the `/SHARE` option, is unrestricted sharing access to open files.

In sharing mode, if you issue the `net use` command to display your network drives, an `/SH` appears after the drive letter of the shared file system.

Use the /MUSTSHARE or /MS switch to specify that you *must* mount the NFS drive in sharing mode. When you specify /MUSTSHARE or /MS, the system mounts the NFS drive in sharing mode, just as it did when you used the /SH switch.

For detailed information about how to use the sharing switches and for examples, see the net use command in Chapter 8.

Read-Only File Systems

There is one other switch for the net use command: the READONLY (/RO) switch, which provides read-only access to a network drive. All three net use switches affect the way in which the system enforces locking and sharing for the entire drive.

You can protect yourself against inadvertent errors by using the /READONLY switch when you issue the net use command. This switch prevents you from changing or erasing files on the drive you specify. You can also activate the /READONLY switch from the nfsconf program. You might want to use the /READONLY switch, for example, when you mount a file system that contains another user's files.

Backing Up and Restoring Files

5

Backing up your files is an essential part of working with a PC. Before you installed PC-NFS, you regularly backed up your files from your PC's hard disk or diskettes to a set of backup diskettes. Now, PC-NFS helps you simplify backing up and restoring your files.

Choosing a Backup Method

Before you implement any of the backup and restoration methods described in this chapter, check with your system administrator. Be sure that the method you choose complements the backup procedures already set up for your network.

If you have PC-NFS LifeLine, you'll probably want to use **LifeLine Backup**. LifeLine Backup allows you to back up your files from your PC onto a tape or disk on your network.

If you do not have LifeLine Backup, you might want to reread the descriptions of the DOS `BACKUP` and `RESTORE` commands. You need to decide on a backup method by choosing:

- Which files to keep on an NFS server's disk
- Which files to keep on your PC's hard disk or diskettes

If you keep all of your files on an NFS server's disk, you might be able to rely on your system administrator to back up your files. In this case, you wouldn't need to do any backups at all. If you don't want to rely on your system administrator, you can back up your files from the NFS server's disk to a local hard disk or diskettes. See the Application Notes chapter of *Installing PC-NFS, A Guide to the User and System Administrator* for further recommendations on where to keep PC application programs and data.

If you rely on your system administrator to back up your files, find out how often the backup is done. If the time period is too long for comfort, choose another method for backing up your files. Choose the backup method that provides you with the best combination of convenience and safety.

If you back up your files from an NFS server, the method you use can vary, depending on whether you are using

- A diskette-only PC
- A PC that has a hard disk

If you are using a diskette-only PC, your program and data files probably reside in directories on an NFS server. In this case, you might rely completely on your system administrator to back up your files. However, if the network or the NFS server malfunctions, you won't be able to use your programs or data files.

To guard against this inconvenience, you could copy your most useful files from the NFS server to a set of diskettes. If there are problems on your network, you could use these diskettes to continue working.

If you are using a PC that has a hard disk, you might also want to back up your working files from the NFS server to your local hard disk. Since some PC application programs are copy-protected, you may not be able to store them on an NFS server, anyway. Note that you can still have simultaneous access to data files on NFS servers. For more information about how using PC-NFS affects application program copy protection schemes, see the Troubleshooting chapter of *Installing PC-NFS, A Guide to the User and System Administrator.*

The rest of this chapter gives you instructions for the following file backup and restoration procedures:

- Backing up to an NFS server
- Backing up to local hard disk or diskettes
- Restoring files to local hard disk or diskettes
- Restoring files to an NFS server

The examples in this chapter assume you are using a local hard disk on drive C of your PC. If you are using a diskette instead, substitute the appropriate drive letter when you try these commands.

Backing Up Files to an NFS Server

As a new PC-NFS user, you might choose to work on your PC as usual, while backing up your files to an NFS server. You can use the DOS BACKUP command to copy files from your local hard disk or diskette to an NFS server's disk.

Let's assume that you have a hard disk on drive C on your PC and a mounted file system on drive D. You can back up the files to drive D by typing:

```
C>BACKUP *.* D:
C>
```

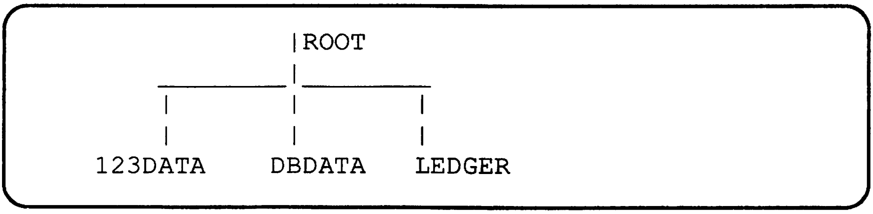
The preceding command line causes DOS to copy all files in the current directory on drive C to a directory on drive D called BACKUP. Under PC-NFS, the BACKUP command works as it normally does when you back up one hard disk to another. As usual, DOS creates a directory called BACKUP into which it copies the files you specify. The files you just copied to drive D (an NFS server's disk) can now be taken care of by network backup procedures.

Remember: You can't use files in a BACKUP directory directly with PC applications. You must first restore these files with the DOS RESTORE command.

The amount of disk space available to you on an NFS server determines how many backup files you can store there. If several users have access to the same NFS server, your system administrator might limit the amount of disk space each person can use. Typically, you would then back up only those files that change from day to day.

Once again, let's assume you have a local hard disk on drive C that includes several subdirectories. Figure 5-1 shows a possible directory structure for drive C.

Figure 5-1 *Backup Example*



If your current directory is the root directory (\), you could back up subdirectories 123DATA, DBDATA, and LEDGER by typing:

```
C>BACKUP 123DATA\*.*  D:
C>BACKUP DBDATA\*.*  D:/A
C>BACKUP LEDGER\*.*  D:/A
C>
```

The first command line causes DOS to erase all of the files in the BACKUP directory on drive D before backing up the files in the 123DATA subdirectory. Cleaning out the BACKUP directory conserves your disk space on the NFS server. The /A option in the second and third command lines then instructs DOS to append the files in subdirectories DBDATA and LEDGER to the BACKUP directory on drive D. After DOS executes the three command lines, the BACKUP directory contains all of the files in subdirectories 123DATA, DBDATA, and LEDGER.

There are ways to simplify maintaining a BACKUP directory on an NFS server. Once you have done your first backup to the NFS server, you can do subsequent backups using either the /M or the /D:mm-dd-yy options. The /M option backs up only those files that you modified since the last time you did a backup. The /D:mm-dd-yy option backs up only those files that you modified on or after the date you specify.

At the end of each work day, you could back up all of the files that you modified by typing:

```
C>BACKUP 123DATA\*.* D:/A/M
C>BACKUP DBDATA\*.* D:/A/M
C>BACKUP LEDGER\*.* D:/A/M
C>
```

When you use both the /A and /M options, DOS appends the modified files to the end of the BACKUP directory on drive D. However, backing up files like this eventually fills your BACKUP directory with obsolete file data. Therefore, you should update your BACKUP directory at least once each week by typing:

```
C>BACKUP *.* D:
C>
```

The preceding command line causes DOS to erase all files in the current BACKUP directory on drive D and then to back up all files in and below your current directory. Updating your BACKUP directory in this way conserves disk space on the NFS server that stores your backup files.

Backing Up Files to Your Local Hard Disk

If you use an NFS server's disk to store files that you use often, network backup procedures take care of these files. You might, however, want to back up your files from the NFS server to local hard disk or diskette for peace of mind. This way, you always have access to your files, even if the network is malfunctioning.

Let's assume that you have a file system mounted on drive D and you want to back up the contents of directory D:\DATA to a hard disk on drive C. You'd type:

```
D>BACKUP \DATA\*.* C:
D>
```

If you did daily backups, you could use the preceding command line at the beginning of each week. Then, you could use the /M option on subsequent days to back up any files modified since the last backup.

```
D>BACKUP \DATA\*.* C:/M
D>
```

Combining these two command lines means that you could never lose more than one day's work in the event of an error or accident.

Restoring Files to Your Local Hard Disk

Backing up files on an NFS server provides you with extra protection for your file data. If for some reason you damage or accidentally erase your local copy of a file, you can use the DOS `RESTORE` command to copy the file from the NFS server that contains your backups. Remember: The `RESTORE` command works only on those files that you copied to the NFS server with the `BACKUP` command.

Let's assume you backed up some files to an NFS file system on drive `D`. You accidentally erased a file called `SCHEDULES` and want to restore it to your local directory `C:\123DATA`. You'd type:

```
D>RESTORE D: C:\123DATA\SCHEDULES
D>
```

You could also specify a group of files that you want to restore by including global file name characters in the file path name. For more information about global file name characters, see your *DOS User's Manual.*

If you wanted to restore the entire contents of directory `123DATA`, you'd type:

```
D>RESTORE D: C:\123DATA\*.*
D>
```

Restoring Files to an NFS Server

If you backed up files from an NFS server to a local hard disk or diskette, you might need to restore them back to the NFS server at some point.

Let's assume that you have mounted an NFS file system on drive `D` and want to restore a file called `UPDATES` from a diskette on drive `A`. You'd type:

```
C>RESTORE A: D:UPDATES
C>
```

Again, before you implement any of the backup and restoration methods described in this chapter, check with your system administrator.

Using Printers

6

PC-NFS provides you with the following print services:

- Shared access to high-quality and high-speed print devices
- High-speed spooling of print data
- Incorporation of new printing technology with familiar PC applications

This chapter describes the print services available on Sun NFS servers. If you want to know whether similar services are available on your system, see your system administrator. If you *are* the system administrator, see the configuration chapters in *Installing PC-NFS, A Guide to the User and System Administrator* for information about server-side print service management.

Using Your Local Printer

PC-NFS does not interfere with the operation of the printer attached to your PC. Keep in mind that your printer cannot be made *directly* available to other users on the network. If, however, other users make their files available to you by placing them on the file server, you can print them locally.

Using Remote Printers

PC-NFS allows you to configure any DOS logical print device (`LPT1`, `LPT2`, or `LPT3`) to refer to a remote network printer. Thereafter, output is channeled from a holding area in your PC (called a *buffer*) to a holding area on the printer server (called a *spool area*) and ultimately to the printer itself. Depending on your application, this flow is initiated in one of three ways:

- You use the print command of your application program that automatically sends data to the printer.
- You use the DOS `COPY` command to copy a file to `LPT`*n*, directly or in a batch file
- You use the PC-NFS `net print` command

Of these methods, the third is the most efficient. PC-NFS incorporates support for the other two mechanisms to ensure that any application that can run with a local printer can also run with a remote printer (unless the program controls the printer hardware directly).

Determining Your Printing Options

The printing options at your site depend on your applications and on your print server(s). In general, files to be printed can be ASCII or files specially encoded for a specific type of device. Any printer can print out an ASCII file. To print specially encoded files, such as PostScript files:

- The print server must support the type of printer required, and
- PC-NFS must be properly set up during installation

The next section, which describes the Sun print server environment, discusses these requirements.

Sun Print Servers. The Sun print servers work with a range of printers. One of the most versatile of these is the Sun LaserWriter™ laser printer, which provides the following options:

- When you send an ASCII file to the Sun LaserWriter, it prints the unadorned file in typewriter (pica) font.
- When you send a file encoded for a Diablo® model 630 printer (if the proper initialization has occurred to the Sun LaserWriter, it creates a facsimile of the output that a Diablo would produce, including bold, underlining, and so on.
- When sent a PostScript™-encoded file, the Sun LaserWriter formats and prints the file to the full resolution available from the Sun LaserWriter.

Redirecting DOS Printer Output

You can redirect printer output from local to remote printers with the following command line:

```
C>net use printdevice:  \\hostname\printername
```

where *printdevicecan* be LPT1, LPT2 or LPT3. Be sure to include a colon and a space after the *printdevice* argument. You can substitute prn for LPT1. You can also use the Resources command of the nfsconf program to redirect printer output.

The command line above redirects files that you send to the DOS device *printdevice* to the device *printername* on the NFS server *hostname.* Note that the remote printer *printername* need not be physically connected to the NFS server *hostname.*

PC-NFS lets you redirect up to three printers at a time. These three printers correspond to the DOS devices LPT1, LPT2, and LPT3. The net use command allows you access to different printers (possibly on different NFS servers) or access to the same printer in different ways. For more information about the latter, see the section *Using a Sun LaserWriter*, later in this chapter.

When you have redirected printer output, any DOS operation that prints data is affected. Examples include:

- Pressing the PrtSc (Print Screen) key.
- Copying a file to a DOS print device using the DOS COPY command.
- Typing the LIST command when using BASIC.
- Loading a spreadsheet into Lotus 1-2-3® and selecting Print, Printer and Go.
- Loading a document into SSI WordPerfect™, pressing Shift F7 and typing 1.

A Caution on DOS PRINT. Now that you have PC-NFS, don't use the DOS PRINT command unless you are sending files to your PC's local printer. The PRINT command can interfere with PC-NFS operations. Always use the net print command when printing files on a remote printer.

If You Don't Know the Name of a Remote Printer

If you don't know the name of the remote printer you want to access, try using `lp` for *printername*. (`lp` is likely to exist on every Sun server with an attached printer of any sort.) You could also ask your system administrator to determine the name for you.

Redirected Output

Where your redirected output prints depends on how your printer is configured. By default, when you redirect printer output to a remote printer, NFS server *hostname* collects the print data in a spool file until:

- A DOS program exits
- No printing occurs for five minutes (called a *time-out*)
- You use the printer Hot key
- You explicitly initiate printing with the `net print` command

When any of these events occurs, the NFS server sends the contents of the spool file to *printername.*

For more information specifically about the printer Hot key, and generally about configuring printers, see the section on printer configuration in *Installing PC-NFS, A Guide to the User and System Administrator.*

Printing a UNIX file under DOS. The following command can be used to print a UNIX formatted file directly from DOS:

```
C>unix2dos unixfile LPTn
```

Spooling Modes and Escaping to DOS. Some PC applications, such as Lotus 1-2-3 (Version 2) and SSI WordPerfect, let you "escape to DOS" without having to terminate your program. If you use this feature and enter `exit` after the DOS prompt, PC-NFS normally prints your spooled files because it thinks you exited the application. This can cause your file to be broken into parts during printing.

If you don't want to break up a print job, don't escape to DOS while running a PC application unless you've selected manual print initiation.

Printing with net print

Once you have redirected your local printer's output, you print a text file on a remote printer by *copying* it. For example, to print a file called yourfile, you enter:

```
C>COPY yourfile LPT1
```

This is a somewhat slow process for a large file, however. For more time-efficient print operations, use the PC-NFS net print command:

```
C>net print yourfile
```

If you don't specify a print device, PC-NFS assumes you want your file to print on LPT1. You can send your file to LPT2 or LPT3 by including LPT*n*: at the end of the preceding command line (you need the colon).

You can issue multiple file name arguments in the net print command, and you can use DOS wildcard characters (* and ?) in the file name arguments. When you print multiple files, your output is a single print job with form feeds separating the files.

Note that when the * character is used alone as a net print argument, the * represents spooled files, not a wildcard for all files. See the section, *Using Remote Printers*, earlier in this chapter for more information.

Remember: The net print command works with redirected print devices only. If you have not redirected the output for printer LPT*n*:, you'll see the following NFS system message:

```
NFS041F : You have not done a net use LPTn: \\host\printername.
```

Using a Sun LaserWriter

If you have configured a Sun LaserWriter, the following sections may be of interest to you.

PC-NFS printing is optimized for use with the Sun LaserWriter, available in a package that includes TranScript™ software from Adobe Systems, compatible with the UNIX operating system.

PC-NFS supports four ways of printing on the Sun LaserWriter printer:

- As a line printer emulator
- As a PostScript printer
- As a Diablo-630 printer emulator
- Raw (unfiltered) — see the following section on "Printing to Non-Sun Printers"

Sun LaserWriter as a Line Printer. When you simply redirect printer output with the `net use` command, the Sun LaserWriter prints your data in *pstext* mode using a simplified fixed-width font. This format is the same one you get when you copy a file to an NFS server and print it with `lpr(1)`.

Sun LaserWriter as a PostScript Printer. If you have a PC application that can generate PostScript commands, you can use a remote Sun LaserWriter with that application.

Sun LaserWriter as Diablo-630. You can use Diablo-630 printer emulation on a Sun LaserWriter with applications such as word processing packages that can take advantage of a high-quality printer with proportional fonts and micropositioning. To use Diablo-630 emulation, first configure your application for this type of printer. Then use the PC-NFS configuration program, `nfsconf`, to select `Diablo 630` mode and configure the emulated printer.

Please refer to the documentation for TranScript's ps630 command for restrictions on Diablo-630 emulation.

Printing to Non-Sun Printers

To maintain compatibility with the Sun TranScript software for the Sun LaserWriter, PC-NFS normally filters out unexpected control characters in the print stream. If your printer uses these control characters (e.g. HP LaserJet™), you should select RAW mode, which passes all data unfiltered to the print spooler.

Using Network Applications

7

This chapter discusses the rsh command and describes how to use the telnet terminal emulator utility. This chapter contains:

- *Introducing* telnet – An overview of telnet and its uses.
- *An Alternative to* telnet*: The* rsh *Command* – A brief description of rsh, which allows the user to execute a single command on a remote machine. The rsh program is an alternative to the telnet program, which facilitates a remote login session.
- *Using* telnet – Shows you how to begin and end a telnet session. This section also describes the telnet Hot key feature, which lets you move easily between your DOS environment and a session on a host system.
- *Using* telnet *Menus* – Describes the two types of menus you can use to select telnet options: initialization menus and control menus.
- *Storing Previous* telnet *Settings* – Describes how telnet stores your previous settings in the session file.
- *Example: Mailing a DOS File* – Shows you how to send a file from a directory on your PC to a user on a host system.

You can find information on telnet troubleshooting in *Installing PC-NFS, A Guide to the User and System Administrator*, in the chapter "Troubleshooting PC-NFS".

Introducing telnet

The telnet program allows a PC running DOS to behave like a terminal attached to a Telnet server. A Telnet server is a computer that supports telnet using Transmission Control Protocol/Internet Protocol (TCP/IP). Sun hosts support these two protocols, as do many UNIX systems and other systems available from a number of vendors.

When you use `telnet`, your PC's keyboard and screen emulate the popular DEC™ VT100 terminal. PC-NFS `telnet` also allows you to transfer files between your PC and other systems on the network. Note that for a simple file transfer your best choice is to use the `ftp` command. See Chapter 8 for information on using `ftp`.

An Alternative to `telnet`: The `rsh` Command

The `telnet` program allows you to log into a remote host system and work in that environment. If you only need to issue a single command, however, you might prefer to use the `rsh` command. This command has the following format:

```
rsh host [ -l username ] command
```

host is the name of the system to which you establish a connection, and *command* is the remote command you want to execute. Generally `rsh` terminates once *command* has executed.

For a more detailed description of the `rsh` command, see Chapter 8.

Using `telnet`

This section shows you how to begin and end a `telnet` session.

You can use `telnet` to communicate with another system on the network after you boot your PC and start up PC-NFS.

> **Note: Check with your system administrator to make sure that the `telnetd` daemon is running on the server system.**

Beginning a `telnet` Session

In its simplest form, `telnet` can be started from the command line without using menus. The command line syntax is:

```
C>telnet host [/b] [/k character]
```

where *host* is COM1, COM2, or a server hostname, and *character* specifies the Hot key character, as described below.

If your PC-NFS uses the same com port as you specify with the `telnet` command line, you must specify *host* on the command line.

When you use `telnet com1` or `telnet com2` to establish a serial connection, `telnet` initially displays a menu that lets you specify parameters such as baud rate and parity.

The /b switch places telnet directly in the background without popping up any menus. This is useful when telnet is started from a batch file.

The /k switch allows you to change the Hot key sequence to something other than Alt T. (As described in the section *Using the telnet Hot Key Feature*, later in this chapter, the telnet Hot key feature lets you change quickly from your DOS environment to a session on a host system.)

The allowable Hot key characters are **A–Z, 0–9,** and **F1–F10**. You can precede these characters with one or more of the keywords Alt, Ctrl and Shift. Separate each Hot key character with a "+". For example, to change the Hot key to Ctrl F10, invoke telnet as follows:

```
C>telnet hostname /k Ctrl+F10
```

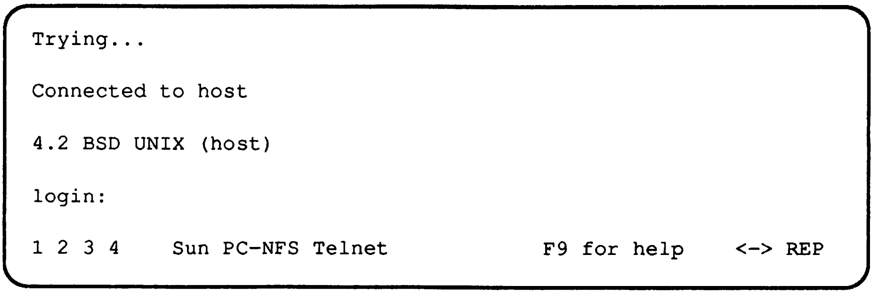
To change the Hot key to Alt Shift Q, the command is:

```
C>telnet hostname /k Alt+Shift+q
```

If you don't specify a host name, telnet displays a series of initialization menus. The section *Initialization Menus* later in this chapter describes selecting options from these menus.

Logging in to a Host System

Once telnet connects your PC to a host, your screen looks like this:



```
Trying...

Connected to host

4.2 BSD UNIX (host)

login:

1 2 3 4    Sun PC-NFS Telnet              F9 for help    <-> REP
```

To log in, enter your login name and password as prompted.

> **This section assumes that you want to log in to a Sun host. For information about logging in to a non-Sun host, see the manual from the appropriate vendor.**

The last line in the preceding display is the Telnet Status Line. The four digits on the left side of this status line correspond to the four Light Emitting Diodes (LEDs) on a VT100. If you run an application program that changes the settings for any of these LEDs, telnet highlights the corresponding digit(s) on the Telnet Status Line.

The F9 key on your keyboard is the *Help key*. This key gives you access to:

- The `telnet` Help Menu
- Other `telnet` control menus

The section *Control Menus* later in this chapter describes the options on these menus.

Specifying the VT100 Terminal Type

After you log in to a Sun host, specify that you want your Sun host to emulate a VT100 terminal by typing:

```
%setenv TERM vt100
```

If you add this command line to the `.cshrc` file in your UNIX home directory, you automatically specify the VT100 terminal type each time you log in.

If you are logging in to a non-UNIX host, ask your system administrator how to specify the VT100 terminal type.

VT100 Terminal Emulation

The VT100 terminal keyboard is laid out somewhat differently from your PC keyboard. The following table explains what PC keys you should use to generate some of the VT100 characters:

Table 7-1 *VT100 Character Generation*

VT100	**IBM PC Keys**	
Characters	Both Shift and NumLock or Neither	Either Shift or NumLock But Not Both
	Numeric Keypad	
0		0
1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
- (hyphen)	*	*
, (comma)	-	-
Enter	+	+
PF1	F1	Fl
PF2	F2	F2
PF3	F3	F3
PF4	F4	F4
	Cursor Control	
← (left)	4	
→ (right)	6	
↑ (up)	8	
↓ (down)	2	
	Input Control	
Delete	Del	
Line Feed	Ctrl J	

Changing Connections

You can establish a new connection after making `telnet` resident, without having to remove `telnet` from the system. However, it is only possible to change from a network connection to another network connection or from a serial connection to another serial connection. It is not possible to switch from a serial connection to an Ethernet connection or vice versa. (This restriction exists to keep the size of resident `telnet` to a minimum.)

To change the *type* of connection, you must remove `telnet` from memory and start it up again.

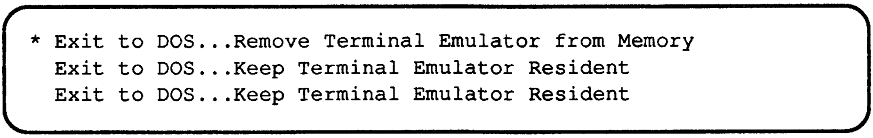
Ending a `telnet` Session

When you end a `telnet` session and return to your DOS environment, you have the options to:

- Leave `telnet` resident in memory, or
- Deactivate `telnet` completely

The following sections discuss the relative merits of these two options and help you evaluate which is best for you.

Leaving `telnet` Resident. If you want to end your `telnet` session and leave `telnet` resident in memory, press the F10 key. Your screen now displays the `Exit Menu` with the following options:



```
* Exit to DOS...Remove Terminal Emulator from Memory
  Exit to DOS...Keep Terminal Emulator Resident
  Exit to DOS...Keep Terminal Emulator Resident
```

The `Exit Menu` also displays the amount of PC memory available if `telnet` is resident.

Use the arrow keys to select the second item and press the Enter key. Your screen then displays the DOS prompt.

Once `telnet` is resident, it remains in memory until you reboot your PC or you Hot key back into `telnet` and remove the program from memory. If your PC has a small memory and `telnet` is resident, system performance might deteriorate.

Once `telnet` is resident, you can use the `telnet` *Hot key* feature. This feature lets you return to your DOS environment *without* ending your `telnet` session. The section *Using the telnet Hot Key Feature* later in this chapter shows you how to do this.

Deactivating telnet Completely. To end your `telnet` session and deactivate `telnet` completely, display the `Exit Menu` by pressing the F10 key. Then, select the first item:

```
Exit to DOS...Remove Terminal Emulator from Memory
```

`telnet` can only be removed from memory after being made resident if no other programs are loaded above it. If the menu item above does not appear, some memory above `telnet` is in use. It's possible that DOS is temporarily using this memory. If so, you can free the memory by entering an external command such as `ls`.

To remove `telnet` from memory after having made it resident, press the Hot key (the default is Alt T) and select the menu item:

```
Exit to DOS...Remove Terminal Emulator from Memory
```

Be sure to use the Hot key and not the `telnet` command to get to the `Exit Menu`. If you use the `telnet` command, the `telnet.exe` program runs above the resident part of `telnet`. This in turn prevents you from removing `telnet` from memory.

Using the telnet Hot Key Feature

When `telnet` is resident, you can use the `telnet` Hot key feature to easily move between your DOS environment and your session on a host system.

The `telnet` Hot key feature lets you use key combinations to change environments:

- Press the F10 key and select the second menu item:
  `Exit to DOS...Keep Terminal Emulator Resident.`
- Press the Alt and T keys together to return to your session on the host system.

The following example shows how the Hot key feature lets you change environments while preserving your work on both systems.

Hot Key Example. Assume you want to log in to the host `sun1`. From your DOS environment, enter:

```
C>telnet sun1
```

After you log in to `sun1`, display the `Exit menu` by pressing the F10 key. Then, make `telnet` resident as described in the section *Ending a telnet Session.* Making `telnet` resident enables the Hot key feature and returns you to your DOS environment.

You might now begin a GWBASIC® session like this:

```
C>GWBASIC

GW-BASIC 2.02
(C) Microsoft 1983, 1984
Compatibility Software GW-BASIC V2.02
Copyright (c) 1984 by Phoenix Software Associates LTD.

62179 bytes free
ok
PRINT ABS (7*(-5))
35
ok
```

At this point, if you press the Alt and T keys, `telnet` returns you to host `sun1` and reprints your last screen from that environment. When you finish working on `sun1`, you can return to your DOS environment by pressing the F10 key followed by the Enter key. The `telnet` program then redisplays your last DOS screen:

```
C>GWBASIC

GW-BASIC 2.02
(C) Microsoft 1983, 1984
Compatibility Software GW-BASIC V2.02
Copyright (c) 1984 by Phoenix Software Associates LTD.

62179 bytes free
ok
PRINT ABS (7*(-5))
35
ok
```

During the rest of your `telnet` session, you can change environments whenever you choose.

Changing the `telnet` Hot Key. You cannot change `telnet`'s Hot key with the `/k` switch without specifying the name of a host. (If you do, `telnet` will complain that it can't find a host named "`/k`".) This means that you cannot use the menu-driven host selection feature and change the Hot key at the same time. You must use the following form:

```
C>telnet hostname /k Ctrl+f1
```

Using telnet Menus

The telnet program includes a number of menus from which you can set various telnet options. These options allow you to dynamically adjust many of the parameters of a telnet session.

There are two types of telnet menus:

- Initialization menus
- Control menus

Initialization menus automatically appear when you enter the telnet command without a host name. Initialization menus allow you to specify:

- What type of terminal emulation you want to use
- How your PC is connected to the host system
- Which host system you want to connect to

Control menus are available once you begin a telnet session. These menus let you select a variety of telnet options, such as setting tabs and displaying the telnet Status Line.

The following sections describe each of the telnet initialization and control menus.

Initialization Menus

Initialization menus let you select telnet options before telnet becomes active. There are three initialization menus:

- Terminal Emulation Menu
- Connection Type Menu
- Select a Host Menu

To choose an item from an initialization menu, select the item with the arrow keys or the Tab key and then press the Enter key.

To return to your DOS environment without selecting any item on an initialization menu, press the F10 key.

> **You can use the Exit menu to return to your DOS environment only when the telnet connection is active.**

When your screen displays the initialization menus, the telnet connection is not yet active. Under these circumstances, you cannot display or select items from the Exit Menu.

The following sections describe the options you can select from each initialization menu.

Terminal Emulation Menu. The Terminal Emulation Menu lets you select the type of terminal emulation you want:

```
A. VT100
B. none
```

When you enter the telnet command without an accompanying host name (and telnet is not already resident), the Terminal Emulation Menu appears on your screen. Unless you are using a special application that doesn't require terminal emulation, select item A.

Connection Type Menu. After you select a terminal emulation type, the Connection Type Menu appears on your screen. This menu lets you specify how your PC is connected to the network:

```
A. network
B. com1
C. com2
```

network refers to a standard Ethernet communications line. com1 and com2 refer to the serial communications ports on your PC.

Generally, you should select item A. If the Ethernet is not functioning properly, however, and there is a physical communications line between your PC and the host system, you have an alternative means of connection: you can select item B or item C and use telnet through one of your PC's serial ports. In this case, if your PC only has one serial port, select item B.

Select a Host Menu. When you select item A from the Connection Type Menu, the Select a Host Menu appears on your screen.

The first time you use telnet, the Select a Host Menu displays only:

```
A. other
```

If you select this item, your screen looks like this:

```
A. other            Enter the hostname:
```

Enter the name of the host system you want to connect to. After you enter a valid host name, `telnet` asks:

```
Do you want to add this host to the known list of hosts?
```

If you select `yes`, `telnet` adds this name to the `Telnet Host List`. `telnet` then uses the `Telnet Host List` to build the next `Select a Host Menu`, which can list up to 20 host names. (If you want to remove some of the host names from this list, you can edit the `TNHOSTS` file in your `\NFS` directory, usually on drive `C` of your PC.)

Your screen now displays the login prompt for the host system you named.

Control Menus

Control menus let you select `telnet` options while `telnet` is active and you are connected to a host system. There are four control menus:

- `Help Menu`
- `File Menu`
- `Change Parameters Menu`
- `Telnet Command Menu`

There are two ways to display a control menu:

- Press a function key or key combination, or
- Display the `Help Menu` and select the control menu you want.

To select an item from a control menu, highlight the item with the arrow keys or the Tab key and then press the Enter key.

To return to your DOS environment from a control menu, use the F10 key.

To return to your session on the host system, press the same function key that you used to display the menu.

> **When `telnet` is active, the F1–F4 keys emulate PF1–PF4. Under these conditions, the F6, F8, F9, and F10 function keys control `telnet` operation; function keys F5 and F7 are inactive.**

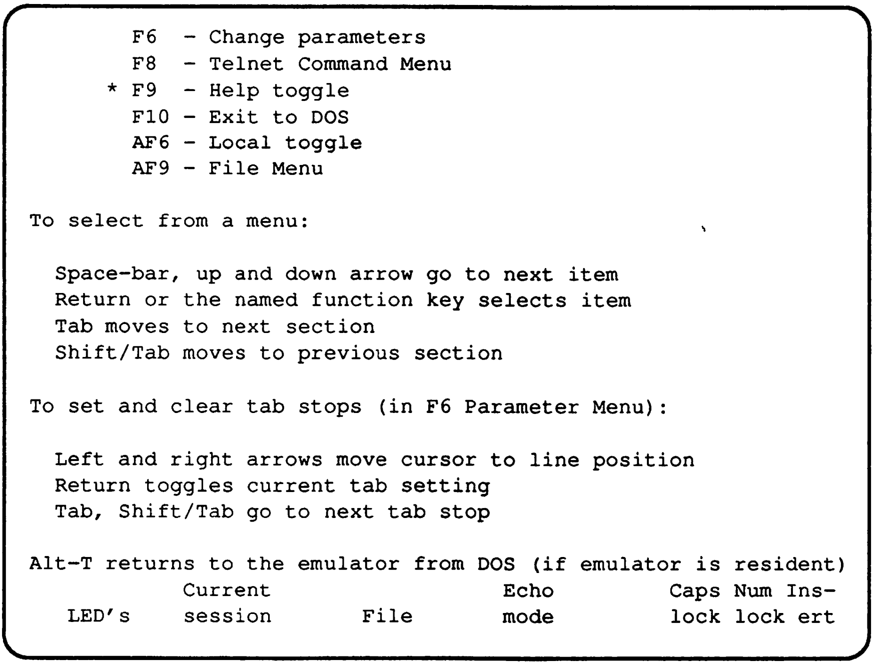
The following sections describe the options you can select from each control menu.

Help Menu

The Help Menu tells you:

- Which keys cause telnet to display the other control menus
- How to select and scroll through menu items

To display the Help Menu, press the F9 key. Your screen now looks like this:



```
        F6   - Change parameters
        F8   - Telnet Command Menu
      * F9   - Help toggle
        F10  - Exit to DOS
        AF6  - Local toggle
        AF9  - File Menu

To select from a menu:

  Space-bar, up and down arrow go to next item
  Return or the named function key selects item
  Tab moves to next section
  Shift/Tab moves to previous section

To set and clear tab stops (in F6 Parameter Menu):

  Left and right arrows move cursor to line position
  Return toggles current tab setting
  Tab, Shift/Tab go to next tab stop

Alt-T returns to the emulator from DOS (if emulator is resident)
            Current                   Echo        Caps Num Ins-
   LED's    session       File        mode        lock lock ert
```

Note that "AFn" denotes the key combination Alt and a function key (F6 or F9). Press the Alt and function keys simultaneously. AF6 is used to toggle you to and from Local mode as described on the next page.

You can display any other control menu by selecting the appropriate item from the Help Menu.

File Menu

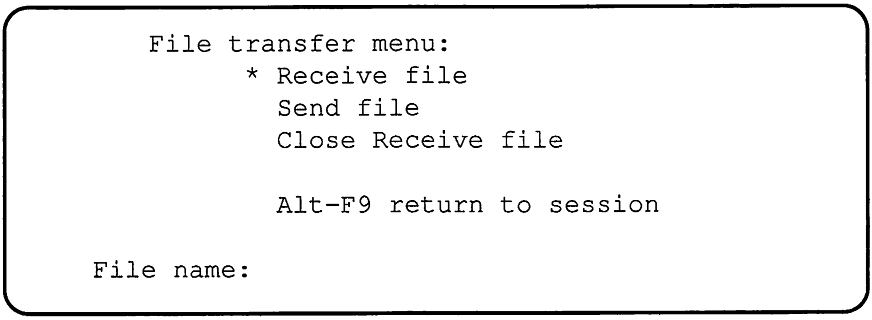
The File Menu includes the following items:

- Receive file – creates a file on your PC that records the output from your session on the host system.
- Send file – sends a file from your PC to the host system.

- `Close Receive file` – stops recording session output in the previously created `Receive file`.

To display the `File Menu`, press the Alt and F9 keys simultaneously.

The `File Menu` allows you to transfer information between your PC and your `telnet` host. The File Menu looks like this:



```
          File transfer menu:
                * Receive file
                  Send file
                  Close Receive file

                  Alt-F9 return to session

     File name:
```

To select an option, use the up and down arrow keys to position the asterisk next to your selection.

To send a file to your `telnet` host:

1. Begin your `telnet` session.
2. On the remote system, prepare to receive a file. For example, create a new file by typing the following `cat` command:
   `cat > destinationfile`.
3. Press the Alt and F9 keys simultaneously.
4. Select `Send file`.
5. Enter the file name.
6. Close the receiving file on the remote system. For example, if you were receiving the file using `cat`, use a Ctrl D to close the file.

To receive a file from your `telnet` host:

1. On the remote system, prepare to send a file to `stdout`. For example, type **`cat < sourcefile`**. Do *not* press the Enter key.
2. Press the Alt and F9 keys simultaneously.
3. Select `Receive file`.
4. Enter the file name.
5. On the remote system, press the Enter key.
6. When you are ready to close the file, press the Alt and F9 keys simultaneously.
7. Select `Close Receive file`.

Note that as long as your receive file is open, the status line at the bottom of your telnet screen displays the name of your receive file in your status line in the format >filename.

You may prefer to use the ftp command to send and receive files between your PC and remote systems. See Chapter 8 for a discussion of ftp.

Change Parameters Menu

The Change Parameters Menu lets you control the following terminal display characteristics. To display the Change Parameters Menu, press the F6 key.

- Terminal Mode – offers you three choices: No Echo, Echo, and Local.

  Generally, you should select No Echo mode. This is because most host systems automatically echo the characters you enter on your PC screen. However, some hosts do require you to select Echo mode. Ask your system administrator which option is appropriate for you.

  Local mode prevents telnet from sending the characters you enter to the host system. You can reset your terminal to Local mode at any time by pressing the Alt and F6 keys simultaneously.

- Status Line – controls whether or not the bottom line of your screen displays the telnet Status Line. The section *Logging in to a Host System* describes the information that this status line displays.

- Auto Line Feed – causes remote hosts that don't echo line feeds to your terminal to do so. If you are connecting to a non-UNIX operating system host, you might need to set this option.

- Video Retrace Wait – lets PCs requiring Video Retrace Wait to function correctly. If your PC requires Video Retrace Wait and you *don't* set this option, your screen displays "snow."

  Note: When Video Retrace Wait is set to "on," you might notice a slower response from your PC.

- Xon/Xoff – enables and disables the use of Ctrl-S and Ctrl-Q for controlling the flow of your screen display. The default is Xon/Xoff disabled.

- End of Line Sequence – allows you to change the sequence of characters sent to a host to indicate the end of an input line. On some non-Sun systems telnet daemons do not recognize the default <CR><NULL> as an end of line indicator (in some systems the null is not passed up to the daemon). In most cases, these systems will recognize the alternative <CR><LF>.

- `Reset Terminal Setup` – allows you to reset to standard VT-100 terminal characteristics. This is similar to the Reset key on the VT-100 keyboard. This feature is most useful when the terminal emulator is in an unusual state due to binary data sent to it.
- `Tabs setting` – lets you set tabs on your screen. Use the Tab key to move the cursor to the bottom of your screen, where the tab settings are displayed. Then, use the arrow keys to move from column to column, and the Enter key to select the desired tab settings. You can also use the Enter key to cancel existing tab settings.

Serial Port Use

If you are using a serial port on your PC to connect to a host system, the `Change Parameters Menu` displays two additional items:

- `Baud Rate` – sets the speed at which data moves across the communications line.
- `Parity` – sets the type of error checking that the communications line uses.

You must configure your PC's serial port so that it matches your connection with the host system. If you aren't sure how to set these parameters, check with your system administrator.

The `telnet` Command Menu

This menu is available only when using `telnet` over the Ethernet.

The `Telnet Command Menu` provides the following `telnet` control functions:

- `Re-open Connection` – reconnects your PC to the host system. Select this item if your connection to the host system was broken by an unusual event, such as a problem on the network.
- `Display Status` – displays the name of the host system to which you are connected.
- `Toggle Append extra Line-Feed Mode` – causes your terminal to transmit a line feed whenever you press the Enter key.
- `Exit Command Menu` – returns you to your session on the host system.

To display the `Telnet Command Menu`, press the F8 key.

Storing Previous `telnet` Settings

The first time you use `telnet`, the program creates a file called `em.ses`. This file contains the `telnet` settings you select from the initialization and control menus (for example, the baud rate).

When you begin subsequent sessions, `telnet`:

- Uses the contents of the `em.ses` file as default settings for your current session.
- Replaces existing `em.ses` settings with any new options you select during your current session.

The `em.ses` file thus spares you the need to select the same menu options at the beginning of every `telnet` session.

If your `telnet` host crashes or your `telnet` session has an abnormal termination, delete your `em.ses` file before you restart `telnet`. It is also a good idea to delete your `em.ses` file when you upgrade to a new release of PC-NFS.

The `em.ses` File Directory. By default, `telnet` stores the `em.ses` file on the drive where PC-NFS is installed in the directory `\NFS`.

If you want to store `em.ses` elsewhere, use the following command line from DOS:

```
    C>SET em.ses=path
```

where *path* is the location of the directory in which you want `em.ses` to reside.

If your PC doesn't have a directory called `\NFS` and you don't specify a location for the `em.ses` file, `telnet` creates `em.ses` in your current directory.

Note: Never edit your **`em.ses`** file directly. Let **`telnet`** create and maintain this file.

You don't need to take any active steps to keep your `em.ses` file up-to-date; `telnet` does it all for you. And if you accidentally delete your `em.ses` file, `telnet` creates a new file at the beginning of your next session.

When `telnet` Is Not Functioning Normally

If `telnet` isn't functioning normally, the contents of your `em.ses` file might somehow have been altered. If this has happened, your screen may display spurious characters instead of the text file you were expecting.

If this type of problem arises, end your current `telnet` session and delete your `em.ses` file. When you begin your next `telnet` session, the program will create a new file.

Example: Mailing a DOS File

The following example shows you how to send a file from your PC to a user on a host system.

Assume that you have a file called `stuff` (in DOS format) in directory `current` on drive `C` of your PC. You want to mail this file to a user called "Dave" on host `sun1`.

1. To convert the file to UNIX format, enter:

   ```
   C>dos2unix stuff stuff.unx
   ```

2. To connect to host `sun1`, enter:

   ```
   C>telnet sun1
   ```

3. Log in to `sun1`.
4. Enter:

   ```
   %mail dave
   ```

   The host system now prompts you to enter a mail message:

   ```
   Subject:
   ```

5. Instead of entering a mail message, press the Alt and F9 keys simultaneously and select the `Send file` item from the `File Menu`. `telnet` then displays the following prompt:

   ```
   File name:
   ```

   Now enter the following path:

   ```
   File name: C:\current\stuff.unx
   ```

   and press the Enter key.

6. Press the Ctrl and D keys simultaneously to mail the file `stuff` to user Dave.

Commands Reference

8

This section contains an alphabetical listing of the commands provided with PC-NFS.

Each command is described by the following sections:

The **command name** and **description** briefly describe the command's actions.

Command Format summarizes the format and available options:

Items in `typewriter font` are to be entered exactly as shown.

Items in *italics* are placeholders to be filled in with your own values.

Items in [square brackets] are optional.

Items separated by vertical bars "... | ... | ..." are *mutually exclusive*: you use only one of them at a time.

Options describe the effects that the different option switches have on the command.

Note that most PC-NFS commands also accept the `-v` option switch, which displays the command's version number.

Also, when several option switches are *not* mutually exclusive, they may be run together on the command line. For example, the command `ls -a -l` could also be entered as `ls -al`.

Examples show some samples of how to use the commands. The portions which are entered by you are shown in **`bold`**.

Notes describe potential problems that could occur with the command in some situations.

See Also refers to related sections and commands in this and other manuals.

arp — Address Resolution Protocol

The arp command displays Internet and Ethernet addresses, and also can set and remove Ethernet addresses from PC-NFS databases. These addresses are used by the Address Resolution Protocol (ARP) to route data to specific machines.

Command Format

arp *hostname* | -a | -d [*hostname* | *]
or
arp -s *hostname ether_addr* | -f *filename*

Options

hostname
Displays the Internet address and the corresponding Ethernet address for the named host.

-a
Displays the Internet address and the corresponding Ethernet address for every known host.

-d [*hostname* | *]
Removes the named host from the table of addresses, thereby forcing PC-NFS to reacquire its address via arp the next time the host is referenced. If *hostname* is *, the entire table of addresses is emptied.

-s *hostname ether_addr*
Set the Ethernet address for *hostname* to *ether_addr*.

-f *filename*
Reads the file *filename* and sets the Ethernet address of each host specified in the file to the specified address. This is the same as executing a series of arp -s commands, one per host. The file *filename* consists of lines, one per host, of the form:

```
hostname ether_addr
```

Examples

To display the Internet and Ethernet address of the host duffer, enter

```
C>arp duffer
duffer (191.9.215.4) at 8:0:20:1:b2:7c
```

To display the addresses for all hosts known to your system, enter:

```
C>arp -a
corona (191.9.216.46) at 8:0:20:1:b3:54
duffer (191.9.215.4) at 8:0:20:1:b2:7c
mudpie (191.29.215.84) at 8:0:20:1:b2:7c
```

An example of a file you might use with the `-f` *filename* option is

```
C>type arpfile
corona 8:0:20:1:b3:54
mudpie 8:0:20:1:b2:7c
```

Notes

The `-s`, `-f`, and `-d` commands should be used with caution, since setting an Ethernet address incorrectly, or removing one or all of the addresses, can result in your PC being unable to communicate with a host whose Ethernet address is incorrect or not found.

chmod Change NFS File Protection

The chmod command allows you to change the protection of NFS files on a UNIX server. This command does not work with local files on the PC.

The chmod command enables you to allow or deny access to your files on an owner, group, and other user basis. You can use chmod to make a file read-only, such as an executable program file (.EXE), or accessible only by you.

Command Format

chmod [ugoa][+-=][rwxstugo] *filename ...*
or
chmod *absolute-mode filename ...*

where:

[ugoa][+-=][rwxstugo] is the symbolic file protection mode to use, as described below.

absolute-mode is an octal number (described below) that specifies the file protection mode to use.

filename ... is one or more file names that have their protection changed by the command.

To change a file's protection using chmod, you must change the *mode* of the file. The mode you specify can be *symbolic* or *absolute*, meaning that you can specify it using a series of letters (symbolic) or using an octal number (absolute).

Symbolic Mode. The *symbolic* mode is made up of three pieces:

1. The *user class* [ugoa], for which the protection is to be changed. This is any or all of the letters u, g, o, and a, which correspond to four classes of users:

u	owner	the owner, or user, that created the file
g	group	the users in the same group as the owner
o	others	other users on the system
a	all	all of the above. This class is equivalent to ugo.

   If none of these is specified, a (all) is used as the default.

2. The *type of change* [+-=] to the protection for the user class (a single symbol):

+	add	add the permission to the user class
-	remove	remove the permission from the user class

= assign — assign *exactly* the specified permissions to the class. Those permissions that are specified are added, and any not specified are removed.

3. The *permissions* `[rwxstugo]` that are changed or assigned:

   `r` read — permission to read the contents of the file

   `w` write — permission to write to the file

   `x` execute — permission to execute a file if it is executable or a UNIX shell script; or, if the file is a directory, permission to search in the directory

   `s` set id — set the owner or group id — useful only if the user is the owner (`u`) or in the owner's group (`g`) and the file is on a system running under UNIX

   `t` sticky — enables you to speed up the execution of a file that is frequently used by causing the system to keep the file in memory or in the swap area between executions

   `u`, `g`, or `o` — indicate that the permissions are to be taken from the current mode for that user class

The symbolic mode may also be made up of several of these pieces, separated by commas. For example, to give yourself read, write, and execute permission, and all others only read permission, you could use `u=rwx,go=r`.

Absolute Mode. Instead of using letters to designate who is able to access the file or execute it, you can use the *absolute mode*. The absolute mode is an octal number, made up of values as follows:

4000	set user id on execution
2000	set group id on execution
1000	sticky bit
0400	read by owner (`u=r`)
0200	write by owner (`u=w`)
0100	execute (search in directory) by owner (`u=x`)
0700	read, write, execute (search) by owner (`u=rwx`)
0070	read, write, execute (search) by group (`g=rwx`)
0007	read, write, execute (search) by others (`o=rwx`)

Examples

To make a program called dodo.exe executable using a symbolic mode, enter:

```
D>chmod +x dodo.exe
```

The command example above changes the modes (chmod) of the file by making it executable (+x).

To see what permissions are set on file, type: **ls -l**

```
C>ls -l
total 194
-rwxrwxrwx 1 tupjohn   28434 Aug 12 12:52 dodo
drwxrwxrwx 2 tupjohn     512 Aug 14 14:16 ORIGINALS
-rw-rw-rw- 1 tupjohn      70 Aug 12 13:02 form
```

The access permissions are represented by the letters in the first column. They appear in groups of three letters in the order of: owner's permissions, group permissions, and others' permissions. A hyphen (–) before a letter indicates that the mode is not permitted.

To make the program dodo execute-only for group and others, and read/write/execute for the owner using the absolute mode, enter:

```
D>chmod 0711 dodo
```

See Also

ls command, later in this chapter.

net umask command, later in this chapter.

UNIX chmod(1) documentation.

Discussion of NFS file attributes in Chapter 3.

connect Connect PC-NFS Through Modem

The `connect` command establishes a connection from PC-NFS to a network server through a dial-up serial line. You must configure PC-NFS to use a serial connector, using the `nfsconf` program, to use this command.

Command Format

`connect [-d] `*scriptname*

where:

scriptname is the name of a serial connection script. The script should have been set up using the configuration program, and contains modem and network information.

Usually, the *scriptname* is the name of the host. However, you may find it convenient to set up different scripts for the same host; for example, one for local calling, and one for long distance.

Options

`-d` Display a detailed trace of activity during the connection process. This can be useful for troubleshooting if any problems arise when you're establishing a connection.

See Also

`hangup` command, later in this chapter.

Discussion of serial configuration in *Installing PC-NFS, A Guide to the User and System Administrator.*

dos2unix — Force File Into UNIX Format

The dos2unix command changes a file from DOS file format into UNIX format. It strips all carriage returns that precede a new line (line feed) in a file, and strips any end-of-file marker from the file.

Command Format

`dos2unix` [`-b` | `-u` | `-l`] [`-f`] [*input-file* [*output-file*]]

where:

input-file, if not present, defaults to standard input.

output-file, if not present, defaults to standard output.

You can use the DOS I/O redirection facility to direct input from batch files or other programs and to direct the output to files.

Options

Note that the `-b`, `-u`, and `-l` options are mutually exclusive.

`-b` (Binary) Converts a DOS file containing 8-bit data to UNIX file format. For example, Wordstar creates files using all 8 bits. The `dos2unix` command normally truncates all characters to 7-bit ASCII.

`-u` Converts text to upper case.

`-l` Converts text to lower case. UNIX commands and files are *case sensitive* and are generally lowercase. For example, some MS-DOS C compilers accept upper and lower case text. UNIX C compilers do not accept keywords in uppercase.

`-f` (Force) Prevents removal of any extra carriage returns. The `dos2unix` command then forces the translation of only a single carriage-return/line-feed pair of characters into a line-feed.

dos2unix normally removes any extra carriage-return characters at the end of a line, and leaves each line with only the proper line ending for UNIX. The `-f` option prevents removal of these extra carriage returns. Normally it should not be needed but it is available so that complete invertibility of files with unusual end-of-line sequences can be preserved if ever required.

Examples

The following two examples have the same result: the input file, infile, is converted from DOS format to UNIX format and written to the output file, outfile.

```
C>dos2unix infile outfile
```

```
C>dos2unix <infile >outfile
```

See Also

unix2dos command, later in this chapter.

ftp File Transfer Program

`ftp` is the user interface to the ARPANET standard File Transfer Protocol (FTP). It is an *interactive* program that enables you to transfer files to and from a remote file system. It prompts you for a command, acts on it, and prompts again for another command.

Command Format

`ftp [-b] [-d] [-g] [-i]` [*hostname*] [*port*]

where:

hostname is the name of a host with an FTP server. If a host name is specified, `ftp` tries to establish a connection with that host immediately upon starting. If `ftp` successfully connects to the host, it automatically issues a `user` command and prompts you for your user name and password.

port, if specified, is the number of a port on which `ftp` attempts to contact an FTP server on the *host*.

After the `ftp` program starts, and whether it connected to a host or not, the program enters its command interpreter. When `ftp` is awaiting your commands, it displays the following prompt:

```
ftp>
```

To leave `ftp`, enter `bye` or `quit`.

Options

`-b` Show all responses from the remote server, as well as report on data transfer statistics. This option is turned on by default if `ftp` is running interactively with its input coming from the user's terminal. This is the same as the `verbose` command, described below.

`-d` Enable display of debugging messages. This is described below under the `debug` command.

`-g` Turn off expansion of wildcard characters in file names (*globbing*). This is the same as the `glob` command, described below.

`-i` Turn off interactive prompting during multiple file transfers. This is the same as the `prompt` command, described below.

As a default, when you issue one of the multiple file transfer commands — `mdelete`, `mget`, or `mput` — `ftp` prompts you for confirmation of the transfer for each file being transferred.

ftp Commands

Command arguments that have embedded spaces can be enclosed by quotation (") marks. If a required command argument is not specified, `ftp` prompts for that argument.

Files specified as arguments to `ftp` commands are processed according to the following rules:

1. If *globbing* (global expansion of file names) is turned on, file names with wildcard characters (`*` and `?`) are expanded according to DOS rules if the expansion is local, or according to the remote host's operating system if the expansion is remote.
2. If globbing is turned off, `ftp` interprets all wildcard characters literally. For example, `put  foo*` looks for a file called `foo*` in the current directory.

When specifying remote files, you must use the remote host operating system format for file names. When you specify a filename for the `get` command, for instance, the filename may not be a legal DOS file name. When the file is transferred, `ftp` prompts you for a legal DOS file name.

Single File Transfer Commands. These commands operate on a single file.

`append` *dos-file* [*remote-file*]

Append a DOS file *(dos-file)* to a file on the remote machine. If *remote-file* is left unspecified, the *dos-file* name is applied to the remote file. The current settings for *representation type*, *file structure*, and *transfer mode* are used while transferring the file.

You should not attempt to append a file to itself. To avoid doing this inadvertently, refrain from creating local directories with the same name as remote directories.

`delete` *remote-file*

Delete the file *remote-file* on the remote machine.

`get` *remote-file* [*dos-file*]

Same as `recv` command; see the `recv` command.

`recv` *remote-file* [*dos-file*]

Retrieve the *remote-file* and store it under the name *dos-file*. If *dos-file* is not specified, the *remote-file* name is used to name the new local file. The current settings for *representation type*, *file structure*, and *transfer mode* are used while transferring the file.

Be careful: If you omit the *dos-file* parameter for a `get` command, the command uses the *remote-file* parameter as the name for the *dos-file*. The `ftp` program interprets the file name literally and creates an identical file name, including path name. For example: `get /usr/foo/file1` creates a file called "`file1`" in the DOS directory `\usr\foo`.

`put` *dos-file* [*remote-file*]

Same as `send` command; see the `send` command.

`send` *dos-file* [*remote-file*]

Store the DOS file *dos-file* on the remote machine. If *remote-file* is not specified, the *dos-file* name is used to name the remote file. The current settings for *representation type*, *file structure*, and *transfer mode* are used while transferring the file.

Be careful: If you omit the *remote-file* parameter for a `put` command, the command uses the *dos-file* parameter as the name for the *remote-file*. The `ftp` program interprets the file name literally and creates an identical file name, including path name. For example: `put c:\foo\file1` creates a file called `c:/foo/file1` on a UNIX system.

`rename` *original-name new-name*

Rename the remote file *original-name* to have the name *new-name.*

`ftp` translates ASCII files, when the representation type is ASCII, in the following ways:

- If you use a `get` or a `recv` to bring an ASCII file to your PC, it arrives in DOS format.
- If you send an ASCII file to a Sun from your PC, it arrives in UNIX format

You can change these defaults with the transfer control commands. See the "Transfer Control Commands" section later in this chapter for more information.

Multiple File Transfer Commands. The multiple file transfer commands take one or more file names as arguments, and performs the transfer on all of them. If wildcard expansion is turned on (the `glob` command), then file names with wildcards expand into a list of matching file names. Otherwise, the wildcard characters do not expand, but are used as-is.

Files are transferred (for `mget` and `mput`) into the current directory, local or remote), using the same name for the file in both the remote and local directory.

`mdelete` *remote-files ...*

Delete the *remote-files* on the remote machine.

`mget` *remote-files ...*

Expand wildcards in the list of *remote-files...* on the remote machine (if globbing is turned on), and do a `get` for each file name thus produced. Files are transferred into the local working directory.

`mput` *dos-files ...*

Expand wildcards in the list of *dos-files* (if globbing is turned on), and do a `put` for each file name thus produced.

Remote Directory Commands. These commands enable you:

- to establish your working directory on the remote machine
- to display file names on the remote machine
- to create and to delete directories (if the permissions allow that).

Note that remote file names and remote directories must conform to the remote host's operating system format when using these commands.

`cd` *remote-directory*

Change the working directory on the remote machine to *remote-directory.*

`dir` [*remote-directory* [*dos-file*]]

Produce a listing of the directory contents for the directory *remote-directory* on the remote machine, and, optionally, place the output in *dos-file*. If no *remote-directory* is specified, the current working directory on the remote machine is used. If no *dos-file* is specified, or if it is –, the listing is displayed on the terminal.

`ls` [*remote-directory* [*dos-file*]]

Produce an abbreviated listing of the directory contents for the directory *remote-directory* on the remote machine, and, optionally, place the output in *dos-file*. If no *remote-directory* is specified, the current working directory on the remote machine is used. If no *dos-file* is specified, or if it is –, the listing is displayed on the terminal.

`mdir` [*remote-files* [*dos-file*]]

Like `dir`, except multiple *remote-files* can be specified, and if globbing is turned on, wildcards can be used.

`mkdir` *remote-directory-name*

Create a directory named *remote-directory-name* on the remote machine.

`mls` [*remote-files dos-file*]

Like `ls`, except multiple *remote-files* can be specified, and if globbing is turned on, wildcards can be used.

`pwd`

Display the name of the current working directory on the remote machine.

`rmdir` *remote-directory*

Delete the directory *remote-directory* on the remote machine. If this cannot be done for some reason, such as the directory still contains some files, an error message displays.

Connection Commands. These commands allow you to establish and terminate connections to a remote host.

`bye`

Same as `quit` command, see the `quit` command.

`quit`

Terminate the ftp session with the remote server by disconnecting from it (like the `close` command), and exit `ftp`.

`close`

Terminate the ftp session with the remote server by disconnecting from it, and return to `ftp`'s command interpreter for another command. This allows you to start another session with another host.

`open` *host* [*port*]

Establish a connection to the specified *host*'s `ftp` server. You can supply an optional *port* number, in which case, `ftp` attempts to contact an `ftp` server using that port on the *host.*

If you specify a *host* name on the command line when you start `ftp`, it automatically tries to connect to that *host* as if your first command were `open` *host.*

`user` [*user-name* [*password* [*account*]]]

Identify yourself to the remote `ftp` server. If you do not specify a *user-name*, `ftp` prompts you for it, with the default being your current PC-NFS log in name (from the `net name` command). If you do not specify a *password* and the server requires it, `ftp` prompts you for it (after disabling local echo). If you do not specify an *account*, and the `ftp` server requires it, the `ftp` prompts you for it.

Transfer Control Commands. These commands allow you to specify the manner in which `ftp` transfers files between your PC and the remote host. You can specify the *representation type* and *subtype*, the *transfer mode*, and *file structure* for file transfers.

The *representation type* is the method used to represent the file data being transferred. It can be one of:

network ASCII
: (the default) The file data is presumed to be in ASCII, structured as lines of characters. This representation involves mapping the line terminators from the source system into `ftp`'s line terminators, and then to those of the target system.

  In addition, the high bit of every ASCII byte is cleared, and only the lower 7 bits are saved. This can cause trouble when transferring certain types of files, such as databases, or some word-processor files. For these, use binary.

binary
: Sometimes called image, and indicates the data is transferred without any change to its file format.

local byte size
: This usually indicates a byte size of 8, which is used to talk to systems running TENEX.

The carriage control format *subtype* of the *representation type* is the method used to represent line printer controls in the file data being transferred. The only *subtype* currently supported is non-print, which means it gets no special handling.

The *transfer mode* is the method used by `ftp` to ship the data over the network. The only *transfer mode* currently supported is stream mode.

The *file structure* is the type of object being transferred by `ftp`. The only file structure currently supported is "file."

`ascii`

Set the *representation type* to network ASCII. This is the default type when `ftp` starts. The is equivalent to the command `type ascii`.

`binary`

Set the *representation type* to binary. The is equivalent to the commands `type binary` or `type image`.

`form` [*format-name*]

Set the carriage control format *subtype* of the *representation type* to *format-name*. The only valid *format-name* is `non-print`.

`mode` [*mode-name*]

Set the *transfer mode* to *mode-name*. The only valid *mode-name* is `stream`.

`sendport`

Toggle the use of `PORT` commands. By default, `ftp` attempts to use a `PORT` command when establishing a connection for each data transfer. If the `PORT` command fails, `ftp` uses the default data port.

When the use of `PORT` commands is disabled, no attempt is made to use `PORT` commands for each data transfer. This is useful for certain ftp implementations that ignore `PORT` commands but incorrectly indicate they've been accepted.

`struct` [*struct-name*]

Set the *file structure* to *struct-name*. The only valid *struct-name* is `file`.

`tenex`

Set the *representation type* to local byte size, which is used to talk to systems running TENEX. The is equivalent to the command `type tenex`.

`type` [*type-name*]

Set the *representation type* to *type-name*. The valid *type-names* are:

`ascii`	for network ASCII
`binary` or `image`	for binary
`tenex`	for local byte size with a byte size of 8 (used to talk to systems running TENEX).

If no type is specified, the current type is printed. The default type is network ASCII.

Local PC Commands. You can use these commands to issue DOS commands on your PC from within `ftp`, and to use an `ftp` command file to run the `ftp` session.

`!` [*dos-command-line*]

Run the *dos-command-line* as a shell command on your PC using DOS. If no *command* is given, invoke the DOS command processor, usually `COMMAND.COM`. To return to `ftp`, enter `exit` after the DOS command prompt.

`lcd` [*dos-directory*]

Change the working directory on your PC under DOS. If no directory is specified, the user's home directory is used. After the command completes, `ftp` displays the current directory *on the current drive.*

Note that this is a directory change on the current drive (or on the specified drive in the *dos-directory* argument). This option does *not* change the current drive under DOS. To change drives, use the `!` command, for example "`! B:`"

`take` [*dos-ftp-command-file*]

Read commands from a local command file. When the command file finishes (assuming it does not terminate `ftp` with a `quit` or `bye`), `ftp` prompts you for the next command.

`ftp` Behavior Commands.

`?` [*ftp-command*]

Same as `help` command, see the `help` command.

`help` [*ftp-command*]

Print a line of help information about the meaning of *command*. If no argument is given, `ftp` prints a list of the known commands.

`bell`

Toggles whether the bell sounds after each file transfer command is completed. Initially the `bell` is OFF.

`debug` [*debug-value*]

Toggles the debugging mode on or off. If the optional *debug-value* is specified, it sets the debugging level. When debugging is on, `ftp` prints each command sent to the remote machine, preceded by an arrow (`-->`). Initially, debugging is OFF, unless you specified the `-d` switch on the command line.

`glob`

Toggles file name wildcard expansion, or "globbing," between on and off, for the `mdelete`, `mget`, `mls`, and `mput` commands. If globbing is turned off, file names are taken literally. By default, globbing is ON.

Expansion of a directory name is likely to be radically different from expansion of the name of an ordinary file. The exact result depends on the remote operating system and `ftp` server, and can be previewed by using the command `mls` *remote-files* `-`.

`hash`

Toggles the printing hash-signs ("#") for each data block transferred. The size of a data block is 1K bytes. By default, this feature is OFF.

`prompt`

Toggles interactive prompting on (or off). Interactive prompting occurs during multiple file transfers (or deletes) to allow the user to retrieve, store or delete files selectively. The default for prompting is ON, unless you specify the `-i` switch on the command line. If prompting is turned OFF, any `mget` or `mput` transfers all files, and any `mdelete` deletes all files.

`remotehelp` [*ftp-command*]

Request help from the remote `ftp` server. If an *ftp-command* is specified it is supplied to the server as well.

`status`

Show the current status of `ftp`.

`verbose`

Toggle verbose mode between on and off. In verbose mode, all responses from the `ftp` server are displayed to the user. In addition, if verbose mode is on, when a file transfer completes, statistics regarding the efficiency of the transfer are reported. By default, verbose mode is ON if `ftp`'s commands are coming from a terminal, and OFF otherwise. The `-b` switch on the command line inverts the default.

Notes

The fptd daemon must be running on the server for ftp to work.

Using ftp to Transfer Files between your PC and Sun. The ftpd daemon running on SunOS 3.2 has some known restrictions that have been corrected in higher releases of the operating system. For SunOS 3.2 the following restrictions apply:

- With globbing on, if you use mdelete with a wildcard, and the match fails, mdelete deletes all of the files in the current directory. For example, if you use mdelete *z and no file exists whose file name matches, all files are deleted. Other multiple commands (mput, mget, mls and mdir) have similar behavior; if the match fails, all files are used. You should avoid using mdelete with this release of the operating system.
- You can only issue one user command from ftp. Subsequent user commands fail with the message "login with user first." If you break a connection using close, and then re-connect using open, however, the user command works once more until the next close.
- If you use the rename command and the rename is unsuccessful, the next command you enter generates the message "command not under-stood." Simply reenter your command.
- The Sun server assumes that ASCII files that it sends are in UNIX format, each line ending with a "LF" (line feed) character. When it sends UNIX files to a DOS system it translates to DOS format which ends each line with "CR-LF" (carriage return - line feed). If you store DOS files on your UNIX system, they are translated even though they should not be, and each line ends with an additional carriage return, i.e. "CR-CR-LF". To avoid this problem, send DOS ASCII files in *binary mode* so they are not translated.

Using ftp with Other Systems. Note that many ftp server implementations do not support experimental operations, such as print working directory. VAX sites running the BBN FTP server appear to ignore the PORT command while indicating compliance; this locks up all file transfers.

hangup Disconnect PC-NFS From Modem

The `hangup` command terminates a connection from PC-NFS to a network server over a dial-up serial line.

Command Format

```
hangup
```

See Also

`connect` command, earlier in this chapter.

Discussion of serial configuration in *Installing PC-NFS, A Guide to the User and System Administrator.*

ls Display File Information

The `ls` command displays information about files. It displays the name, size, creation date, modification time, and access permissions for files.

Command Format

`ls` [`-a`] [`-d`] [`-g`] [`-b` | `-l` | `-u` | `-w`] [*filename* | *directoryname*]...

where:

filename, if present, is a DOS file name or wildcard pattern that selects which files to display information about.

directoryname, if present, is a DOS directory name. The `ls` command displays the file information for all files in that directory (except with the `-d` switch; see the following).

Options

Note that the `-b`, `-l`, `-u` and `-w` options are mutually exclusive.

With no options specified, the `ls` command produces a file-listing similar to the DOS `DIR` command, showing for each file the DOS name, size, and modification date and time. In addition, it shows the UNIX operating system-style owner permissions, and the NFS file name if the DOS name is a *mapped name*.

`-a` Include hidden files in the listing — those with the hidden bit set in DOS, or for which the UNIX `setuid` attribute is set. The `-a` option can be combined with any of the others, e.g. `-al`.

`-d` Display directories as files. Normally `ls` displays the *contents* of a directory when the *directoryname* appears on the command line. If you use the `-d` option, it lists the information about the directory itself. You can combine the `-d` option combine with any of the others, e.g. `-dl`.

`-b` For each file, list the DOS name and NFS name, side by side. This can be used by a program that wants to operate upon both types of names.

`-g` Used with the `-l` option, `-g` lists the group owner for each file.

`-l` Produce a listing very similar in form to the UNIX `ls -l` command, showing for each file the NFS file name, size, protection, modification date, and ownership, one file per line.

`-u` (Unix) Same as the `-l` option.

`-w` Produce a listing that is similar to the DOS `DIR/W` command, which is just the DOS file names in columns, without the volume and directory name.

Examples

`ls` command entered without options.

```
     1        2        3        4        5       6        7

C>ls
.        <DIR>               1-01-80  12:04a  U:rwx
..       <DIR>               1-01-80  12:04a  U:rwx
DOS2UNI   EXE     10086      4-15-86   3:48p  U:rwx
EM        COM     34398      4-16-86  11:48a  U:r-x
3C501     300      3459      4-15-86   3:24p  U:rwx
NATURE               56      4-15-86   2:58p  U:rwx
RICHES               48      4-15-86   2:58p  O:rwx
SPEED~CA            398      1-09-86   2:35p  U:rwx  speed.dial
THOUGHT              75      4-15-86   2:58p  U:rw-
EM        SES       574      1-01-86   2:40a  U:rwx
```

Column	Description
1	DOS file or directory name
2	Extension
3	Size in bytes
4	Creation or modification date
5	Creation or modification time
6	Owner and permissions. U means that you own the file; G means that someone in a group to which you belong owns the file; and O means that someone who does not belong to one of your groups owns the file. R, W, and X indicate the type of permissions you have on the file.
7	NFS name, if different

The –b style produces a listing with two columns. The first column is the filename in DOS format; the second column is the file name in UNIX format. This illustrates the name mapping between DOS and UNIX names.

```
C>ls -b
..            ..
DOS2UNIX.EXE  dos2unix.exe
3C501.300     3c501.300
NATURE        nature
RICHES        riches
SPEED~CA      speed.dial
THOUGHT       thought
EM.SES        em.ses
```

The -l listing is similar to the UNIX command ls -l:

```
     1      2    3      4     5      6     7
```

```
C>ls -l
drwx------  1 jrs          - Jan  1 00:04 .
drwx------  1 jrs          - Jan  1 00:04 ..
-rwx------  1 jrs      10086 Apr 15 15:48 dos2unix.exe
-rwx------  1 jrs       3459 Apr 15 15:24 3c501.300
-rwx------  1 jrs         56 Apr 15 14:58 nature
-rwxrwxrxw  1 jrs         48 Apr 15 14:58 riches
-rwxrwxr--  1 root        33 Apr 13 14:34 speed.dial
-rwx------  1 jrs         75 Apr 15 14:58 thought
-rwx------  1 jrs        574 Jan  1 02:40 em.ses
```

Column	Description
1	Complete list of NFS permissions
2	Number of NFS file links
3	Name of the NFS file owner or uid if name is unknown
3	Size in bytes
4	Creation or modification date, whichever is more recent
5	Creation or modification time, whichever is more recent
7	Full NFS file name

The first column displays the NFS permission settings for the file, in the format: *duuugggooo*, where the *d* is a d for a directory, an l for a link, or a – for a file, and *uuu*, *ggg* and *ooo* are the permissions (r, w, x, or –) for the owner, group and other users, respectively.

The -w listing is similar to the DOS command DIR/W:

```
C>ls -w
3C501    300   NATURE         RICHES         THOUGHT
EM       SES   DOS2UNIX EXE   SPEED~CA
```

See Also

chmod command, earlier in this chapter

See Chapter 3 for a discussion of mapped names.

mv Rename a File

Then mv command changes the name of a file or a directory, and possibly moves it from one directory to another on the same disk.

Command Format

mv [*drive:*]*oldname newname*

where:

[*drive:*]*oldname* is the name of the original file or directory. It may include a path specification, and an optional drive letter *drive:* indicating the network disk on which it is stored.

newname is the new name for the file, or the name of an existing directory into which to move the file. If *newname* contains a path specification different from the location of the original file *oldname*, the file is moved into the new directory as well as being renamed.

Note that you may not specify a drive letter for newname, as mv cannot be used to copy files across drives.

Examples

For example, to change the UNIX filename thisisalongfilename to thisfile, use the command:

```
D>mv thisisalongfilename thisfile
```

To move the file JANDATA to a different directory on the same disk:

```
D>mv E:\DATACOLL\JANDATA \DATACOLL\OLD\JANDATA
```

myeaddr — Display Ethernet Address

This command displays the Ethernet address of your PC.

Command Format

`myeaddr` *controller_type* [*ioport_addr* | *shmem_addr* [*interrupt#*]]

where:

controller_type is the option name of the communications controller on your system, either the **3C501, 3C503, 3C505, 3C523, NIC, NI5010,** or **WD8003E.**

ioport_addr is the I/O port address for 3C501, 3C503, 3C505 or NI5010 controllers. This argument should be used only when the factory settings have been changed from the default settings.

shmem_addr is the shared memory address for NIC and Western Digital controller boards. This argument should be used only when the factory settings have been changed.

interrupt# is the interrupt number for the controller boards. This argument should be used only when the factory settings have been changed.

Examples

```
C>myeaddr 3c501
Your 3c501 Ethernet address is: 2:60:8c:15:90:96

C>
```

See Also

See the Installation section of *Installing PC-NFS, A Guide to the User and System Administrator* for a discussion of how to determine the *ioport_addr*, *shmem_addr*, or *interrupt#* addresses.

net blip Turn Blip On or Off

The net blip command turns the network activity indicator on or off. When it is on, a rectangle (blip) is displayed in the upper right corner of the screen at the start of every NFS remote procedure call, or whenever PC-NFS sends network data. When the call is completed the rectangle is cleared. This is a network activity indicator for the user.

Command Format

```
net blip[on|off]
```

When used without any arguments, the command displays whether blip is on or off.

Examples

```
C>net blip
NFS020I : Blip (on-screen network activity monitor) is enabled.
C>
```

`net join` Join a Drive Letter to a Path

The `net join` command associates (*joins*) a remote drive with an empty directory on another drive. This allows you to use the joined drive as a subdirectory in another file system.

Command Format

`net join` [*drive1* : *path drive2:*]

or

`net join` *drive*: `/d`

where:

drive1 : *path drive2:*

joins *drive2* to the *path* on *drive1*. The drives must be remote (NFS) drives, different drives, and previously specified in `net use` commands/.

Once *drive2* has been joined to *drive1* : \path, you should not access *drive2* directly under its own drive letter. However, it still shows up in a `net use` listing.

drive: `/d`

disconnects *drive* which was previously joined to another by a `net join` command. This can only be done if the *drive* is not in use, which in this case means that there must not be any other file systems joined to the *drive* and that the current directory is not on the *drive*.

When used without any arguments, the command displays the list of drives and paths which have been joined using this command.

Examples

For example, after the commands

```
C>net use e: \\hostone\usr\me
C>net use f: \\hosttwo\usr
C>mkdir f:\sub1
c>net join f:\sub1 e:
```

you should no longer access drive `E` directly. When you change your current drive to `E` you find your current directory displayed as `D:\SUB1`. However you can ask for a directory of drive `E`, and it shows up in a `net use` listing. You can undo the join with the command **`net join e: /d`**.

`net logout` Log User Off Network

This command logs the user off the network and reverses the effects of a `net name` *username* command. The `net logout` command resets the user name to `nobody`, as it was before the user issued any `net name` commands.

Command Format

```
net logout
```

Examples

To see the effect of the `net logout` command, enter the following commands and verify that the `net name` command indicates you are logged in as `nobody`.

```
C>net logout
C>net name
The name of this machine is nv, and its IP address is 196.9.254.18
No subnet mask has been installed.
It is Yellow Pages domain stpaul.mn, served by minni (196.9.267.2)
The authentication server is minni (196.9.267.2)
You are logged in as nobody, with UID -2 and GID -2.
It is FRI JAN 29 21:35:30 1988 EST

C>
```

net name — Perform NFS User Authentication

The `net name` command logs the user into the network, and establishes the user's network-wide user id (*uid*) and group id (*gid*). With no arguments, it displays current network and user information.

Command Format

```
net name [ *  |  username [ * ]  ]
```

where:

username, if specified, is your login name. If you enter an `*`, the command prompts you to supply your username.

You must use `*` in place of the password. The command prompts you to supply the actual password, which is not displayed as you enter it.

If you use the supplied configuration program, it asks you for your user name, and puts a `net name` *your-username* `*` command into the file `\NFS\NETWORK.BAT`, which runs whenever you boot your PC.

When used with no arguments, `net name` displays the following information:

- the name and Internet address of your PC
- the Yellow Pages (YP) domain name and the name and Internet address of any Yellow Pages server
- the name and Internet address of the authentication server
- the name and Internet address of any gateway system
- your user name, user id, and your primary group id
- the ids of secondary groups to which you belong, if any
- the date, time, and time zone

For the `net name` command to work, the `pcnfsd` daemon must be installed and running, normally on the currently selected Yellow Pages server machine. If you have Yellow Pages, PC-NFS looks for the `pcnfsd` server on the Yellow Pages server. If there is no Yellow Pages server machine, the daemon can run on any server. Use the `net ypset` or `net pcnfsd` commands to set the name of the server running the `pcnfsd` daemon.

net pcnet

Run PC-NFS with IBM PC-Network

The `net pcnet` command enables or disables PC-NFS compatibility with IBM PC-Network.

Command Format

```
net pcnet [on|off]
```

Options

When used without any arguments, the command displays whether PC-Network compatibility is enabled or disabled.

Examples

```
C>net pcnet
PC-NET compatibility is OFF.

C>
```

net pcnfsd — Set or Display NFS Authentication Server

The `net pcnfsd` command allows you to set the name of the NFS authentication server or displays the current `pcnfsd` server name.

Command Format

`net pcnfsd` [*hostname*]

where:

hostname, if specified, is the name of the authentication server PC-NFS should use. The authentication server must be running the `pcnfsd` daemon. Unlike `net ypset`, this command does no checking to ensure that *hostname* is an acceptable server.

If you wish to use `net name` later to establish your user identity, you should use this command to identify the system to which the authentication request should be directed.

If you use the supplied configuration program, it asks you for the name of an authentication server, and puts a `net pcnfsd` *hostname* command into the file `\NFS\NETWORK.BAT`, which runs whenever you boot your PC.

When used without any arguments, the command displays the name and Internet address of the current authentication server.

Examples

```
c>net pcnfsd
The authentication server is a-server (191.7.218.2)
```

Notes

When using the `net pcnfsd` command to set the name of the authentication server, it should be done before the `net name` *username* command, and also before any `net use` *printer* commands.

`net print` Print Files on Network Printers

`net print` allows you to print files on network printers. Without an option, `net print` initiates spooling for the redirected printer and displays a message indicating that it has initiated the command.

Command Format

`net print` [* | *filename...*] [*printdevice*:]

where:

filename..., if specified, is a list of DOS file names to be printed on the *printdevice*. The *filenames* can include DOS wildcard characters (* and ?), to print multiple files. When you enter multiple file specifications, `net print` produces a single print job in which the files are separated by form feeds.

*, if used instead of any *filenames*, forces any print files that have been accumulated in the print queue for the specified printer to be printed. You should use this command only if the designated *printdevice* has been configured for manual print initiation mode using your configuration program.

printdevice, if specified, is then name of the DOS printer to which the command is directed. This may be one of `LPT1`, `LPT2`, `LPT3`, or `PRN` (synonymous with `LPT1`). If *printdevice* is not specified, `LPT1` is used. Note that the *printdevice* name must be followed by a colon.

The command `net print` (without any arguments) is equivalent to `net print * LPT1:`. Since `LPT1` is the most commonly used printer, this is a convenient way of flushing spooled data.

Notes

Before you use a remote *printdevice*, you must mount it using the `net use` command or the `nfsconf` program.

See Also

See Chapter 6, *Using Printers*.

net route Set the Current Gateway

`net route` allows you to set the name of the gateway to use. When a gateway is set, if a server's Internet address indicates that it is not on the network local to your PC, packets for the server are directed to that gateway for routing to the appropriate network.

Command Format

`net route` [*hostname* | `/d`]

where:

hostname, if specified, sets the gateway to the host of that name.

`/d` deletes the current gateway setting.

With no arguments, the command `net route` displays the name and Internet address of the currently set gateway.

Examples

```
C>net route
Non-local routing via gateway r-server (191.7.218.7)
```

`net start rdr` & `net stop rdr` — Start or Stop PC-NFS

These two commands start and stop PC-NFS.

Command Format

```
net start rdr [my_pc [ * ]]
```

or

```
net stop rdr
```

where:

my_pc, if specified, is the name to be used for your PC when connected to the network. You *must* supply this name if the network is running without either Yellow Pages or Reverse ARP (RARP), or if you use the `*`, as described below.

`*` means that Reverse ARP (RARP) should not be attempted and that the local file `\NFS\HOSTS` should be read to find out the Internet address for *my_pc*.

If you use the supplied configuration program, it puts a `net start rdr` command into the file `\NFS\NETWORK.BAT`, which runs whenever you boot your PC.

Notes

The command `net stop rdr` produces an error message if you have any remote file systems mounted, or redirected printers.

The command `net stop rdr` does not remove PC-NFS from memory; it only stops it from handling some DOS operations.

net subnet — Set the Subnet Mask

This command sets and displays the subnet mask.

Command Format

```
net subnet [mask | * ]
```

where:

mask, if specified, sets the subnet mask to *mask*. The value for *mask* can be given in hexadecimal (0xffff0000) or dotted decimal (123.45.6.78) notation.

* broadcasts a request for the subnet mask. If some node replies, the value it returns is used to set the mask. If no node replies, an error message is displayed.

With no arguments, the command `net subnet` displays the current subnet mask, and the corresponding interpretation of the PC's Internet address.

Note

As described in the Technical Reference section of *Installing PC-NFS, A Guide to the User and System Administrator*, the subnet mask cannot be set until the PC's Internet address has been established by the `net start rdr` command.

The `net subnet` command should immediately follow the `net start` command.

See Also

See the Technical Reference section of *Installing PC-NFS, A Guide to the User and System Administrator*, for a discussion of the subnet mask.

net umask Set the Default NFS Permissions

The net umask command sets the mask for the default permissions which are used when a new NFS file is created on a UNIX server.

Command Format

`net umask` [*octal_value*]

where:

octal_value, if specified, is the ones-complement of the permission bits that is set when creating a NFS file.

With no arguments, the command `net umask` displays the mask for current default permissions, in octal.

Common values for the mask *octal_value* are:

022	anyone can read, only the owner can write
077	no access for anyone other than the owner
000	unrestricted access to all

The default value after initialization is 000 (unrestricted access).

Examples

```
C>net umask
NFS0231: The current umask is 000 (octal).
```

Note

The octal value is *ones-complement*, that is, it is the *opposite* of the absolute permissions a new file is given. For example, to give new files the permissions `u=rwx, go=rw` (absolute mode 766), use a mask of 011.

See Also

`chmod` command, earlier in this chapter.

`net use` Mount Remote File Systems

The `net use` command mounts a remote file system on a logical drive, associates a network printer with a DOS printer device, or displays the mounted drives and printer associations.

The `net use` command corresponds to the UNIX `mount` facility.

Command Format

`net use` *drive*`:` *\\hostname\path* [`/SHARE` | `/MUSTSHARE` | `/READONLY`]

or

`net use drive: vmshot:vmspath`

or

`net use` *drive*`: /d`

or

`net use` *printdevice*`:` *\\hostname\printername*

where:

drive`:` *\\hostname\path*

mounts a file system on a PC drive letter. In other words, this associates a drive letter on the PC with a directory in the file system on a host (server).

drive is the letter of a PC drive on which to mount the file system. *drive* can be any letter following your last existing disk drive and up to and including the letter `S`. The drives `T`, `U`, and `V` are reserved for remote printers. Note that it must be followed with a colon (`:`).

hostname is the name of the host where the file system is located. *path* can be the name of an exported file system — `\usr`, for instance — or a subdirectory within an exported file system, such as `\usr\staff\geoff`.

An *exported file system* is a portion of a UNIX file system which the system administrator has made available for other machines to mount.

`/MUSTSHARE` or `/MS` mounts the network drive in sharing mode only. If PC-NFS cannot contact the lock manager on the server, the net use fails.

`/READONLY` or `/RO` mounts the network drive in read-only mode so that you cannot modify or delete files on the disk.

`/SHARE` or `/SH` mounts the network drive in sharing mode, if possible. See Chapter 4 for a discussion of sharing.

drive: /d

tells PC-NFS to stop using the disk *drive* that was previously mounted by net use. This corresponds to the UNIX umount command.

printdevice: \*hostname**printername*

associates a DOS printer device *printdevice* with a printer on a remote system. *printdevice* must be one of PRN, LPT1, LPT2, or LPT3 and must be followed by a colon (:). PRN is a synonym for LPT1. *printername* is the printer name (lp , lw , and so on) on the server system; in SunOS and BSD UNIX operating system, this is the name given in the /etc/printcap file on the host *hostname*.

Subsequent net print commands or printer output operations to the *printdevice* write spool files into the server's spool directory. (See Chapter 6 for further details.)

With no arguments, the command net use displays a list of the currently mounted drives and printers.

Two special names are defined in association with the net use command. $YPSERVER refers to the current Yellow Pages server system. $HOME refers to your home directory as defined in the passwd Yellow Pages map. Thus in the case where your home directory is on the current (or only) Yellow Pages server, you can use:

```
net use d: \\$YPSERVER\$HOME
```

to mount your home directory on drive *d*.

Examples

To mount the directory \usr\tjones, located on the host, yourhost on PC drive E, enter the following:

```
C>net use e: \\yourhost\usr\tjones
```

To mount the printer lp on the host que2, enter the following:

```
C>net use LPT1: \\que2\lp
```

To see what drives are mounted, enter `net use`. The system displays the file systems and printers that are mounted.

```
C>net use
Drive Filesystem                   Tsize  Kbytes Used  Avail Capacity
D:    \\yourhost\usr\tjones\doc 8192  87544  85120  2424   97%
E:    \\yourhost\usr\tjones     8192  87544  85120  2424   97%
LPT1: \\que2\lw
```

Notes

Remember to use backslashes instead of slashes when specifying paths to `net use`.

The names $YPSERVER and $HOME must be entirely uppercase.

The drive letters available to you are `A` through `R`. The drive letters `T`, `U`, and `V` are reserved for remote printers. Drive letters `W`, `X`, `Y` and `Z` are not available because certain software packages such as Microsoft WINDOWS® cannot use them.

The disk space number shown by `net use` usually differs from the number shown by the UNIX `df` command on a host system for the same directory. This difference appears for two reasons. First, the directory might be undergoing change. Second, the UNIX operating system reserves disk space for administrator use; this is a configuration parameter that can vary at each site.

To communicate with a VMS server use the alternate syntax:

```
net use d: host:path
```

in which *path* should not include any slash (/) or backslash (\) characters.

See Also

For more information on file sharing and locking, see Chapter 4.

See the sections in *Installing PC-NFS, A Guide to the User and System Administrator* on the configuration program and mounting drives and printers.

`net version` Display PC-NFS Version Number

This command displays the version number of PC-NFS.

Command Format

```
net version
```

Example

```
C>net version
NFS016I: The PC-NFS Version is 3.0.1
```

net ypdomain Set the Yellow Pages Domain

This command allows you to set the name of the Yellow Pages domain to which your PC belongs. This command corresponds to the SunOS `setdomain` command.

Command Format

`net ypdomain` [*domainname*]

where:

domainname sets the Yellow Pages domain to *domainname*.

With no arguments, the command `net ypdomain` displays the name of the current Yellow Pages domain.

If no `net ypdomain` command is given, PC-NFS uses the default of `noname`.

Usually this command is run from your `\NFS\NETWORK.BAT` file at boot time, which is set up during the configuration process. See the discussion of the `NETWORK.BAT` database file in *Installing PC-NFS, A Guide to the User and System Administrator*.

Example

```
C>net ypdomain
NFS0321: The Yellow Pages domain is yp.server.com
```

Notes

After running `net ypdomain` to set the name of the Yellow Pages domain, you should run `net ypset` *hostname*.

See Also

`net ypset` command, later in this chapter.

net ypset — Set the Yellow Pages Server

This command allows you to set the name of the Yellow Pages server and authentication server.

Command Format

`net ypset` [*hostname* | *]

where:

hostname is the name of the server to use as the Yellow Pages server and authentication server.

If *hostname* is not a Yellow Pages server, or if * was specified, broadcast to find a Yellow Pages server. If a Yellow Pages server is found for the domain specified previously by a `net ypdomain` command, then the authentication server name is set to that Yellow Pages server machine. Otherwise, the authentication server is set to *hostname*.

With no arguments, the command `net ypset` displays the name of the current Yellow Pages server.

Usually this command is run from your `\NFS\NETWORK.BAT` file at boot time, which is set up during the configuration process. See the discussion of the `NETWORK.BAT` database file in *Installing PC-NFS, A Guide to the User and System Administrator*.

Example

```
C>net ypset
NFS)0271: Using Yellow Pages server yp.server.com.
```

See Also

`net ypdomain` command, earlier in this chapter.

`netstat` Display Network Statistics

The `netstat` command displays statistics about the network. This is similar to the UNIX command `netstat(8)`.

Command Format

```
netstat [-s][-i]
```

Options

`-s` (the default) Displays summary information about the network protocols.

`-i` Displays network interface statistics.

Examples

```
C>netstat
TCP:
        0 bad header checksums
        3021 packets received
UDP:
        0 bad header checksums
        1762 packets received
        15 NFS packets received
IP:
        0 bad header checksums
        0 unrecognized packets
        1644 dropped packets (no takers)
ARP:
        0 requests received
        0 responses sent
        2 requests sent
        2 responses received
ICMP:
        0 packets received
        0 packets sent to unreachable networks
```

nfsping — Determine If a System Is an NFS Server

The nfsping command determines if a named system is an NFS server.

Command Format

nfsping *hostname*

where:

hostname is the name of the system to test.

Examples

If the system is alive and well the following message is displayed.

```
C>nfsping hostname
hostname is alive.
```

Otherwise, after about 10 seconds, this message is displayed.

```
C>nfsping hostname
No answer from hostname
```

When used in a DOS batch file, `nfsping` returns an `errorlevel` of 1 if the NFS server does not answer. (However, this command does not return a 0 upon success.) You can use this feature with the DOS `if errorlevel` batch facility to determine whether a server is available before continuing batch processing. For example, the batch command file could contain these commands:

```
C>type sample.bat
        nfsping timbuktu
        if errorlevel 1 goto trydc
        net ypset timbuktu
        goto finish
:trydc
        nfsping washdc
        if errorlevel 1 goto noserv
        net ypset washdc
        goto finish
:noserv
        echo NO SERVERS FOUND
        goto finish
:finish
C>
```

This is useful for overnight batch processing.

nfsstat Display NFS Statistics

The `nfsstat` command displays statistics concerning NFS operation. This is similar to the SunOS command `nfsstat`.

Command Format

```
nfsstat [-c|-n|-r|-b]
```

Options

-c (the default) Displays statistics about client RPC and NFS usage.

-n Displays statistics about client NFS usage.

-r Displays statistics about client RPC usage.

-b Displays statistics about client RPC, NFS, and buffer cache usage.

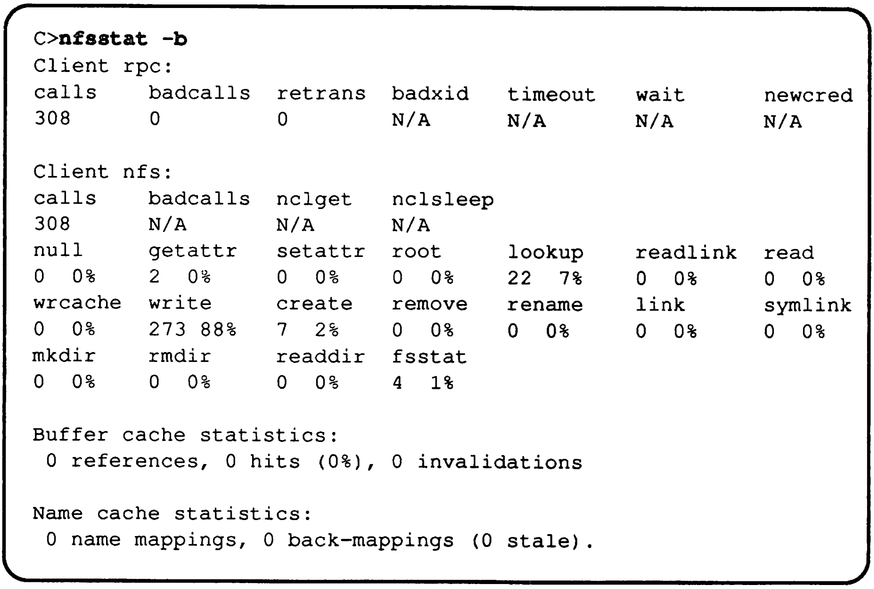
Examples

```
C>nfsstat
Client rpc:
calls     badcalls   retrans    badxid     timeout    wait       newcred
57        0          0          N/A        N/A        N/A        N/A

Client nfs:
calls     badcalls   nclget     nclsleep
57        N/A        N/A        N/A
null      getattr    setattr    root       lookup     readlink   read
0   0%    2   3%     0   0%     0   0%     10 17%     0   0%     0   0%
wrcache   write      create     remove     rename     link       symlink
0   0%    38 66%     3   5%     0   0%     0   0%     0   0%     0   0%
mkdir     rmdir      readdir    fsstat
0   0%    0   0%     0   0%     4   7%
```

```
C>nfsstat -c
Client rpc:
calls    badcalls  retrans  badxid   timeout   wait      newcred
308      0         0        N/A      N/A       N/A       N/A

Client nfs:
calls    badcalls  nclget   nclsleep
308      N/A       N/A      N/A
null     getattr   setattr  root     lookup    readlink  read
0   0%   2   0%    0   0%   0   0%   22   7%   0   0%    0   0%
wrcache  write     create   remove   rename    link      symlink
0   0%   273 88%   7   2%   0   0%   0   0%    0   0%    0   0%
mkdir    rmdir     readdir  fsstat
0   0%   0   0%    0   0%   4   1%
```

```
C>nfsstat -n
Client nfs:
calls    badcalls  nclget   nclsleep
308      N/A       N/A      N/A
null     getattr   setattr  root     lookup    readlink  read
0   0%   2   0%    0   0%   0   0%   22   7%   0   0%    0   0%
wrcache  write     create   remove   rename    link      symlink
0   0%   273 88%   7   2%   0   0%   0   0%    0   0%    0   0%
mkdir    rmdir     readdir  fsstat
0   0%   0   0%    0   0%   4   1%
```

```
C>nfsstat -r
Client rpc:
calls    badcalls  retrans  badxid   timeout   wait      newcred
308      0         0        N/A      N/A       N/A       N/A
```



```
C>nfsstat -b
Client rpc:
calls    badcalls  retrans  badxid   timeout   wait      newcred
308      0         0        N/A      N/A       N/A       N/A

Client nfs:
calls    badcalls  nclget   nclsleep
308      N/A       N/A      N/A
null     getattr   setattr  root     lookup    readlink  read
0  0%    2  0%     0  0%    0  0%    22  7%    0  0%     0  0%
wrcache  write     create   remove   rename    link      symlink
0  0%    273 88%   7  2%    0  0%    0  0%     0  0%     0  0%
mkdir    rmdir     readdir  fsstat
0  0%    0  0%     0  0%    4  1%

Buffer cache statistics:
 0 references, 0 hits (0%), 0 invalidations

Name cache statistics:
 0 name mappings, 0 back-mappings (0 stale).
```

rcp Remote File Copy

The rcp command copies files between systems on the network.

Command Format

`rcp` *sourcefile destinationfile*

or

`rcp [ -r ]` *sourcefile ... destination_directory*

where:

sourcefile is the file name and (optionally) the path name of the file you wish to copy. The files you copy may be either local to your PC (*local names*), or files on other systems (*remote names*). Wildcard characters for local names (* and ? under DOS) are not expanded, but they are expanded for remote names.

destinationfile is the file name (remote or local) to which you wish to copy the *sourcefile*. Wildcard characters are not allowed.

destination_directory, in the second form of the command, specifies a directory (local or remote) into which to copy the *sourcefile*(s). The copied files keep the same file names.

`rcp` handles third party copies, which means that neither the source nor destination files are on your PC.

Options

`-r` If any of the source files are directories, `rcp` copies that directory and all the files and directories within it; in this case the destination must be a directory.

File and Directory Name Formats

Each *file* or *directory* argument is either a remote file name of the form *rhost:path*, or a local file name (containing no " : " characters). Note that this means you *cannot* specify a drive letter for a local file name.

Local Names. You can specify a local file name (or directory name) using either slashes (/) or backslashes (\), i.e. either as `/d/d/f` or as `\d\d\f`. Note that you may not specify a drive letter. To read from or write to a drive, you should make the appropriate drive the current drive under DOS. If the path for the file name does not start at the root, i.e. start with a "\," it is interpreted relative to the current directory on the current drive.

Remote Names. A remote file name (or directory name) has the form *rhost:path,* where *rhost* is the name of the remote system, and *path* is the path to the file or directory. If *path* is not a full path name, it is interpreted relative to your login directory on *rhost.*

You must use slashes (/) in the path for a remote name if it is on a UNIX host.

Then remote name can also take the form *rhost.rname:path,* to use *rname* rather than the current user name when logging into the remote host.

Notes

Remember where you are at all times (putting your host name in your prompt helps with this)!

`rcp` is meant to copy from one host to another; if by some chance you try to copy a file on top of itself, you end up with a severely corrupted file.

`rcp` doesn't detect all cases where the target of a copy might be a file where only a directory should be legal.

`rcp` is confused by any output generated by commands in any of the following files on a remote host: `.login`, `.profile`, or `.cshrc`.

`rcp` doesn't copy ownership, mode, and timestamp values to the new files.

`rcp` does not prompt for passwords; your current local user name must exist on the remote host (unless you used the form *rhost.rname*), and allow remote command execution via `rsh`.

`rcp` requires that the source host have permission to execute commands on the remote host when doing third-party copies.

When you use `rcp` to copy a UNIX file to DOS, `rcp` truncates filenames longer than 8 characters, and truncates extensions longer than 3 characters.

If you attempt to `rcp` a UNIX file with multiple extensions to a DOS directory, `rcp` responds with "`NO SUCH FILE OR DIRECTORY.`"

If you specify a file name with a wildcard in the source files list and specify a file name as the destination instead of specifying a directory, `rcp` copies the last file that matches the wildcard into the specified destination file.

There is no wildcard expansion of source file names which are local names, i.e. files on a DOS disk or an NFS disk mounted by PC-NFS using `net use`.

Example

To copy a file, `monster`, from `lochness` to `champlain`, enter the following:

```
C>rcp lochness:monster champlain:monster
```

If you do not specify a full path name, `rcp` places the file in your log in directory on the remote machine. If you do not have a log in directory, `rcp` returns an error message and does not place the file on the remote machine.

See Also

`ftp` command, earlier in this chapter.

`rsh` command, later in this chapter.

rpcinfo Display RPC Information

The rpcinfo command displays information about remote procedures on other machines. This is similar to the SunOS command rpcinfo(8).

Command Format

rpcinfo -u *hostname program_number version_number*
or
rpcinfo -p *hostname*

Options

-u *hostname program_number version_number*
Determines if program *program_number*, version *version_number* is registered on host *hostname.*

-p *hostname*
Displays a list of all remote programs registered on host *hostname.* It displays the program numbers, version numbers, protocols and port numbers of these programs.

Examples

```
C>rpcinfo -p yamuthah
[program, version, protocol, port]:

[100004, 2, 17, 1027]
[100004, 2, 6, 1024]
[100004, 1, 17, 1027]
[100004, 1, 6, 1024]
[100007, 2, 6, 1025]
[100007, 2, 17, 1035]
```

Notes

The rpcinfo command can be used to see if a remote machine is a Yellow Pages Server (has a program = 100004), if a machine is running NFS (program = 100003), and if it is the latest version (version = 2).

rsh Remote Shell

The rsh command executes a command line on a specified host.

Command Format

rsh *host command*

where:

host is the name of the remote server on which to execute the *command.*

command is the command line to execute.

rsh copies its standard input to the remote command, the standard output of the remote command to its standard output, and the standard error of the remote command to its standard error. Interrupt, quit and terminate signals are propagated to the remote command; rsh normally terminates when the remote command does.

rsh cannot be used with interactive UNIX commands that require user action, such as more. If rsh is used with an interactive UNIX command, some of the output may not be displayed on the PC screen, and you may have to press Ctrl Break to get a new DOS prompt. You can, however, access mail with rsh if you can use mail without prompts.

Special command line characters that are not quoted are interpreted on the local PC, while quoted special characters are interpreted on the remote machine. Thus the command:

```
C>rsh lizard cat lizard.file > tutorial.file
```

writes the remote file lizard.file from the machine called lizard to the file called tutorial.file on your PC.

The command:

```
C>rsh lizard cat lizard.file ">" another.file
```

writes the file lizard.file on the machine called lizard to the file another.file which also resides on the machine called lizard.

Example – Printing with rsh

You can print files from your PC to a printer on a Sun server by piping the output of a DOS command into a UNIX command on the server. rsh allows you to print files on the machine and printer of your choice. Furthermore, you can use the print command of your choice, such as enscript or lpr. For example:

```
C>type dos_file1 | rsh server_a enscript -2r
C>type dos_file2 | rsh server_a lpr -p
```

showmnt Show Exported File Systems

This command displays the exported file systems on a single host or all host systems in the Yellow Pages data base.

Command Format

`showmnt [ -e | -l ]` *hostname*

where:

hostname is the server for which the export list is displayed.

Options

`-e` (the default) Displays the list of exported file systems on the named host.

`-l` Generates the list of exported file systems for every server with file systems mounted on the host *hostname*.

Example

```
c>showmnt your-server
Export list for your-server:
/usr/archive        everyone
/usr/development    everyone
/usr/test           everyone
/usr                everyone
/                   everyone
```

Notes

If the export list is big, PC-NFS's buffer may overflow, giving an error. The only current workaround is to reduce the export list on the server.

unix2dos Force File Into DOS Format

Change a UNIX file into DOS format. This command adds a carriage return before each new line (line feed) character not already preceded by a carriage return..

Command Format

`unix2dos [ -b | -u | -l ] [ -f ] [ -z ]` [*input_file* [*output_file*]]

where:

input file, if not present, defaults to standard input.

output file, if not present, defaults to standard output.

You can use the DOS I/O redirection facility to direct input from batch files or other programs and to direct the output to files.

Options

Note that the `-b`, `-u` and `-l` options are mutually exclusive.

`-b` (Binary) Preserves 8-bit codes; `unix2dos` normally truncates all characters to 7-bit ASCII codes. For example, Wordstar creates files using all 8 bits.

`-u` Converts data to uppercase.

`-l` Converts data to lowercase. UNIX commands and files are *case sensitive* and are generally lowercase. For example, some MS-DOS C compilers accept upper and lower case program. UNIX C compilers do not accept keywords in uppercase.

`-f` Force the addition of a carriage return before a line feed character, even if one was already present.

`dos2unix` normally removes extra carriage returns at the end of a line, and the `-f` option preserves them, only removing the last one at the end of the line. `unix2dos` normally does not add a carriage return before a line feed if there was one present already. The `-f` option causes `unix2dos` to always insert a carriage return, so that complete invertibility of files with unusual end of line sequences can be preserved if ever required.

`-z` Add a **Control-Z** (DOS end-of-file character) to the end of the file, unless one was present already.

Examples

The following two examples have the same effect: the input file, `infile` is converted from DOS format to UNIX format and written to the output file, `outfile`.

```
C>unix2dos infile outfile
```

```
C>unix2dos <infile >outfile
```

See Also

`dos2unix` command, earlier in this chapter.

ypcat

Display Values From Yellow Pages Database

This command displays values in the Yellow Pages database on your current YP server. This is similar to the SunOS command `ypcat(1)`.

Command Format

```
ypcat [-k][-t] mapname
```

or

```
ypcat -x
```

where:

mapname is the name of a Yellow Pages (YP) map, or a map nickname.

Options

`-k` Display the keys as well as the values.

`-t` Do not translate *mapname* as a nickname; use it as the map name.

`-x` Display the map nickname translation table, that is, all nicknames known to the Yellow Pages.

Example

```
C>ypcat -x
Use passwd    for map  passwd.byname.
Use group     for map  group.byname.
Use networks  for map  networks.byaddr.
Use aliases   for map  mail.aliases.
```

See Also

`ypmatch` command, later in this chapter.

ypmatch — Find and Display Entry From Yellow Pages Database

This command searches the Yellow Pages database for entries with a specified key. This is similar to the SunOS command `ypmatch(1)`.

Command Format

`ypmatch [-k][-t]` *key ... mapname*

or

`ypmatch -x`

where:

key is one or more key values which are used to search the map in the Yellow Pages.

mapname is the name of a Yellow Pages (YP) map, or a map nickname.

Options

`-k` Display the keys as well as the values.

`-t` Do not translate *mapname* as a nickname; use it as the map name.

`-x` Display the map nickname translation table, that is, all nicknames known to the Yellow Pages.

Examples

```
C>ypmatch -x
Use passwd    for map  passwd.byname.
Use group     for map  group.byname.
Use networks  for map  networks.byaddr.
Use aliases   for map  mail.aliases.
```

```
C>ypmatch duffer hosts
191.9.215.4    duffer
```

See Also

`ypcat` command, earlier in this chapter.

Glossary

A

Address Resolution Protocol (ARP)
This protocol provides translation of host addresses from Internet addresses to Ethernet addresses.

`ANSI.SYS` driver
This device driver is distributed with the DOS operating system. `ANSI.SYS` must be installed in a PC's system directory so that PC-NFS can properly format screen displays.

application
A computer program that performs a task previously done by another means. Word processing programs, spreadsheet programs, and database programs are examples of applications.

ASCII (American Standard Code for Information Interchange)
A standard code for the representation of alphanumeric information.

authentication
A means to distinguish individual system or network users and to assign these users different privileges.

Authentication server
A machine on the network that has software which authenticates user privileges on the network. For PC-NFS this software is `pcnfsd`.

`AUTOEXEC.BAT` file
This DOS system file can exist on any PC. `AUTOEXEC.BAT` contains a series of command lines that execute when the computer starts up. These commands typically define the operating system environment. This is where the commands go that start up PC-NFS and that set the appropriate parameters for your network environment.

backup
The process of making copies of files, archiving, as protection against the destruction or loss of the original files.

buffer

A temporary storage area in RAM, used typically to hold data being transferred to or from a device.

cache

A section of RAM set aside to hold data that is accessed frequently.

client

A computer that uses resources provided by another machine on the network. Most commonly, these resources are disk storage space and printers. A PC running PC-NFS becomes a client of the network it joins. (See also **server**.)

communications board

A hardware board installed in a PC that provides the Ethernet connection required to run PC-NFS.

compatibility mode

This file sharing mode provides *Deny Write* access if you want to read a file, and *Deny All* (exclusive) access otherwise. By default, DOS opens files in this mode.

`CONFIG.SYS` file

This DOS system file can exist on any PC. `CONFIG.SYS` defines which device drivers to install.

copy protection scheme

A means by which software vendors discourage or prevent users from illegally copying licensed software. PC-NFS itself implements a scheme that detects and publicizes the use of illegal PC-NFS serial numbers.

daemon

A service program that runs continuously on a server system in a UNIX environment. A daemon program generally provides some service or resource to client systems on the network. On Sun systems, daemon programs have a `d` at the end of their program names.

default

For commands that require user input, there is often a value supplied by the program if the user does not enter a value. The value supplied by the program is the default.

directory tree

In the DOS operating system, files are organized in directories. A directory can contain other directories, called subdirectories which also can contain files and directories. This hierarchical structure of directories forms a directory tree.

domain

A named collection of machines on a network.

Ethernet

A type of network that supports high speed communications between machines. The network usually uses coaxial cables.

Ethernet Address

A six-part hexadecimal number in which a colon separates each part (for example, 8:0:20:1:2f:0). This number identifies the Ethernet communications board installed in a PC, and is used to identify the PC as a member of the network.

export

To make a file system available so that one or more other systems can access (mount) it. On Sun servers, this is done by adding an entry in the `/etc/exports` file on the appropriate server system. (See also **mount**, **server**, and **client**.)

file access

The facility that allows a user to work with a remote file as if the file were local. File access differs from file transfer in that it does not create a local copy of the file. (See also **file transfer**.)

file attributes

The qualities that describe a file and the ways in which it can be accessed and used. File attributes include among others, file names, file permissions, and user classes.

file format

The way in which data is represented in a file. This representation varies among different operating systems. When you work with PC-NFS, you can use the `dos2unix` and `unix2dos` programs to convert a DOS file to UNIX format and vice versa.

file locking

A service to prevent simultaneous updates of the same file by different users. PC-NFS supports advisory file locking, which allows you to maintain a consistent database if you enforce advisory file locking for all of your client systems. PC-NFS does not support mandatory file locking, in which the system verifies every file access.

file name mapping

A PC-NFS service by which NFS (UNIX) file names are translated into legal DOS file names. These mapped file names are also converted back to their original NFS forms when necessary.

file permissions

These permissions refer to the three types of file access: read, write, and execute. As the owner of a file, you can grant all or some of these file permissions to user groups.

file system
The organizational structure, composed of directories, subdirectories and files, that an operating system uses to manage data.

file transfer
Copying a file from one computer to another. File transfer differs from file access in that it creates a *second copy* of the file. (See also **file access**.)

ftp (File Transfer Protocol)
A TCP service that transfers files between systems on a network, without regard to the operating systems of the machines involved in the transfer.

gateway system
A system that provides a link between two networks.

group id (gid)
A unique number associated with each group name on a server system. The system assigns your group id to any files that you create.

home directory
A user's default location within an NFS server's file system.

host
The name a network uses to identify a machine on the network.

host name
A unique name that identifies each host machine on the network. You need a database named `\nfs\hosts` on networks without a Yellow Pages server. When this database is present, it contains the unique host names and Internet Protocol (IP) addresses of your PC and, sometimes, NFS servers.

hot-key
A set of keystrokes that allows you to reduce routine procedures to a short sequence. The `nfsconf` program allows you to define a hot-key sequence to send data to a printer. The `telnet` hot-key allows you move easily between your DOS environment and a session on a remote host system.

`inetd` daemon
A service program that runs continuously on a server system in a UNIX environment to invoke connection oriented services.

`install` program
A program that resides on your first PC-NFS distribution diskette that constructs the files necessary for you to run PC-NFS successfully.

Internet Protocols
Computer communication protocols developed by the United States Department of Defense for the scientific and technical fields. They include TCP/IP, Telnet, and FTP.

Internet (IP) Address
A number that uniquely identifies every computer connected to a TCP/IP network.

join
A command to connect a drive to a directory on another drive. The DOS `JOIN` command does not work with NFS files, but `net join` allows you to join one mounted file system to a subdirectory of another.

local
In a network, files, devices, or users that are on your machine.

Locking Services
Sun server software that enables you to protect files from unwanted access.

log in
To gain access to a computer's operating system. This can be accomplished by (1) simply turning on the computer, or (2) completing a clearance procedure such as entering a user name and password.

memory resident
Programs or data loaded into memory, but not saved when the machine is rebooted.

mount
To make a file system on a server available to your PC.

name cache
A list of the last 64 file name mappings created by PC-NFS in converting NFS file names to DOS file names.

`net` program
The PC-NFS program that implements the various network management functions.

Network File System (NFS)
A facility that allows you to share files in a heterogeneous environment of machines, operating systems, and networks.

`nfsconf` program
A program that allows you to configure the operation of PC-NFS.

`\nfs\hosts` file
A file that contains the host name and Internet Protocol (IP) address of your PC, and sometimes, the addresses of NFS servers. You need this file on networks that do not have Yellow Pages or Reverse ARP.

nobody
If you do not specify a user name, the system assigns you this default name. `nobody` is associated with a set of minimum system privileges defined by your system administrator.

noname
The Sun default name for a Yellow Pages domain.

operating system
A collection of programs that monitors the use of the machine and supervises the other programs executed by the machine.

path name
A statement that identifies the position of a file or directory within the tree structure of a file system.

Postscript
A page description language developed by Adobe Systems, Inc. used by a wide range of output devices.

protocol
A set of conventions that govern how machines within a particular network communicate with each other.

RAM
Random Access Memory.

remote
In a network, files, devices, and users not attached to your machine.

remote access
The ability to read and write remote files.

Remote Procedure Call (RPC)
A mechanism that lets a program running on a client machine call a procedure that a remote server executes.

restore
The process of putting backed up files back on a storage device.

Reverse ARP (RARP)
Reverse Address Resolution Protocol allows a machine which is connected to an Ethernet to determine its IP address..

server
A machine that provides resources to other computers on a network. Two examples of servers are: (1) file servers, which provide physical disk storage space, and (2) print servers, which make their attached printers available to other systems. (See also **client**.)

session
A time period defined by the beginning of a program and its termination. A PC-NFS session begins when PC-NFS starts and ends or when the system is shut off or rebooted.

spool
The mechanism that forms a queue of files for a shared resource so that they use the device in an orderly fashion.

subnet
A means of making the Ethernet addresses for small networks more efficient by splitting host numbers into two fields.

subnet mask
The method used to separate the two parts of the host name in a subnet.

SunOS
The operating system, a version of the UNIX operating system, that runs on Sun workstations.

superuser
A special type of system user who can change any other user's file attributes.

system administrator
The person responsible for managing systems on a network, including NFS server systems. The system administrator's responsibilities include updating network databases for new systems and installing network software such as the `pcnfsd` program.

TCP/IP
Transmission Control Protocol/Internet Protocol.

telnet
A protocol for establishing a login session on another computer.

terminal emulator
A program that causes a PC screen and keyboard to act like a video display terminal attached to another computer.

time-out
A pre-defined period of system inactivity, during which the system waits for user or network response. If there is no response before the end of the time-out, the system takes some action.

UNIX
An operating system initially developed by AT&T which is now available from many sources.

user classes
Categories of users used to enforce file protection. The three user classes are *owner*, *group*, and *other*. You can assign file permissions to each user class for each file.

user id
A unique number associated with each user name on a server system. The system assigns your user id to any files that you create.

user name
The character string with which you identify yourself to the system, usually assigned by your system administrator.

VMS
The operating system that runs on Digital Equipment Corporation's VAX family of machines.

Yellow Pages service
The YP services provide access to a set of network databases. The databases contain such information as host names, Internet addresses, and Ethernet addresses. A host machine that provides YP databases is called a YP server.

Index

Symbols

! command (ftp) 80
? command (ftp), *see* help command (ftp)

Numerics

1-2-3, Lotus 43, 44
3C501 89
3C503 89
3C505 89
3C523 89
3Com 89
4.2 BSD (UNIX) 3

A

access
  conflicts 30
  deny write 31
  file 31, 85
  read 31
  remote 9, 12, 13, 47–63, 74–83, 117
access rights, *see* permissions
activity indicator (blip) 14, 90
Address Resolution Protocol 5, 107
Adobe Systems TRANSCRIPT 45
addresses
  Ethernet 15, 66, 89
  Internet 14, 15, 66, 93, 95, 97–99
advisory file locking 30
Alt-T (telnet) 49
append command (ftp) 75
applications
  1-2-3, Lotus 43, 44
  copy protection 36
  defined 6
  PC-Network, IBM 14, 94
  printing 11, 42
  sharing 29
  Windows, Microsoft 103
  WordPerfect, SSI 43, 44
  Wordstar, Micropro 120
ARP 5, 107
arp command (PC-NFS) 15, 66–67
network ASCII 79, 80
ascii command (ftp) 79
ASCII files 42, 76, 83
ATTRIB command (DOS) 25
attributes, file 19
  changing 26
  DOS 20, 25
  hidden, DOS 20
  mapping 19, 25–27
  NFS 21
  PC-NFS 26
  permissions in NFS 21
  read-only, DOS 20
  setuid 85
  user classes in NFS 21
authentication server 14, 15, 93

B

backing up files 35–40
  choosing a method 35
  eliminating the need for 35
  errors 39
  from an NFS server 36
  on a diskette-only PC 36
  on a hard-disk PC 36
  to local hard disk 39
  to NFS server 37–39
  with PC-NFS LifeLine Backup 35

backslashes 103
  path names, DOS 20
BACKUP command (DOS) 37, 38
BACKUP directory 37, 38
behavior commands (ftp) 81–82
bell command (ftp) 81
binary command (ftp) 79
binary representation in ftp 79, 80, 83
blip, *see* activity indicator
bye command (ftp), *see* quit
    command (ftp)
byte size representation in ftp 79, 80
byte-range locking 31

C

cached name handle 23
carriage return 20, 24, 72, 83, 120
case-insensitive 20
case-sensitive 21
cd command (ftp) 77
changing default file permissions 27
changing directories 77
changing file access 14
changing file format 14, 72
changing file name 14
changing file permissions 14, 27, 69
CHKDSK command (DOS) 18
chmod command (PC-NFS) 14, 27, 68–70
  absolute mode 69
  examples 70
  symbolic mode 68
classes, user 21, 30, 68
clients 6
close command (ftp) 78, 83
com ports
  telnet 48
  serial 4
command processor (DOS)
command reference (PC-NFS) 65–123
COMMAND.COM 80
commands, BASIC
  LIST 43
commands, DOS
  ATTRIB 25
  BACKUP 37, 38
  CHKDSK 18
  COPY 18
  DISKCOMP 18
  DISKCOPY 18
  FDISK 18
  FORMAT 18
  JOIN 18
  LABEL 18

commands, DOS, *continued*
  and networked disks 18
  PRINT 43
  RECOVER 18
  RENAME 24
  RESTORE 40
  SET 62
  SUBST 18
  SYS 18
commands, PC-NFS 14-15, 66–123
  arp 15, 66–67
  chmod 14, 27, 68–70
  connect 71
  dos2unix 14, 25, 72
  ftp 74–83
  hangup 84
  ls 14, 24, 26, 85–87
  mv 14, 24, 88
  myeaddr 89
  net blip 14, 90
  net join 18, 91
  net logout 14, 92
  net name 14, 26, 92, 93, 95, 102
  net pcnet 14, 94
  net pcnfsd 14, 93, 95
  net print 14, 42, 43, 45, 96, 102
  net route 14, 97
  net start rdr 14, 98, 99
  net stop rdr 14, 98
  net subnet 14, 99
  net umask 15, 27, 100
  net use 10, 15, 30, 32, 43, 46, 91,
      95, 101–103
  net version 15, 104
  net ypdomain 15, 105
  net ypset 15, 93, 95, 106
  netstat 15, 107
  nfsping 15, 108–109
  nfsstat 15, 110–112
  rcp 12, 18, 19, 113–115
  rpcinfo 15, 116
  rsh 13, 48, 117–118
  showmnt 15, 119
  unix2dos 14, 44, 120–121
  –v option switch 65
  ypcat 15, 122
  ypmatch 15, 123
commands, UNIX
  mount 18
  setenv 50
communications controllers 89
compatibility mode 31

configuration program (nfsconf) 9, 10, 30, 32, 33, 43, 71, 93, 95, 98
conflicting access 30
connect command (PC-NFS) 71
connecting to a remote host (ftp) 78
connection commands (ftp) 78
CONTROL-Z 20
controller boards 89
  I/O port address 89
  shared memory address 89
converting
  file formats 24–25
converting files 14, 72, 120
COPY command (DOS) 12, 18
copy protection 36
copying
  files to a DOS print device 43
  files to a remote printer 45
copying files
  between systems 12, 18–19
  and file formats 19
CR, *see* carriage return
creation date (files) 85

D

daemons
  ftpd 83
  pcnfsd 14, 93, 95
  rshd 12, 13
  telnetd 13, 48
date, network 93
debug command (ftp) 81
default file permissions 26
default permissions 100
default user name 26
delete command (ftp) 75
deleting directories 78
deleting remote files
  delete command (ftp) 75
  mdelete command (ftp) 77
denying file access 31
df command, (UNIX) 103
Diablo
  630 emulation mode 42, 46
DIR command (DOS) 85
dir command (ftp) 77
directories 6
  access to 14
  changing remote 77
  copying nested 19
  displaying information about 85
  home 102
  joining, *see* net join command (PC-NFS)
  list contents 14
directories, *continued*
  moving files between 88
  moving files in 14
  name expansion in ftp 81
  names 20
  permissions for 14
  permissions under NFS 27
  remote commands (ftp) 77–78
  removing remote 78
  renaming 88
  restoring 40
  tree structure 6
    illustration 7
  working 77, 78
disconnect, *see* hangup command (PC-NFS)
disconnecting drives 91
disconnecting sessions (ftp) 78
disk usage 103
DISKCOMP command (DOS) 18
DISKCOPY command (DOS) 18
disks
  backing up files to local hard 39
  restoring files to local hard 40
displaying
  exported file systems 15
  information about files 27
  Internet address 15
  Yellow Pages values 15
domain
  Yellow Pages 105
  noname 105
DOS
  3.1 file sharing functionality 29
  and PC-NFS 4
  command processor 80
  commands from ftp 80–81
  COPY command 12
  copy to LPT mechanism 42
  DIR command 85
  escaping to 44
  escaping to, from ftp 80
  file attributes 25
  file format 24, 72, 76, 83, 120
  file names 20
  file sharing modes 30, 31
  file systems 6
  files 6, 75, 77, 83, 85
    on a remote system 76
  logical device names 43
  and PC-NFS 18
  printer designations 42
  printers 10
  printing UNIX files under 44
  vs. UNIX 24

PC-NFS
    and DOS 18
dos2unix command (PC-NFS) 14, 25, 72
drive letters, available 103
drives
    connecting 91
    disconnecting 91
    joining 91
    logical 101
    network, *see* remote file systems

E

Echo Mode, telnet program 60
EDLIN program 27
em.ses file (telnet) 62
emulation
    Diablo-630 42
    Diablo-630 printer 46
    line printer 46
    terminal 47
    VT100 13
end-of-file, DOS 20
environments
    operating systems 17–27
escaping to DOS 44
    from ftp 80
/etc/printcap file 102
Etherlink boards 89
Ethernet address 15, 66, 89
Ethernet networks 4
executable files
    storage 27
execute permission 21, 68
executing a remote command 13
exiting to DOS, *see* escaping to DOS
export list 119
exported file systems 101, 119
extensions
    executable files, DOS 27
    file name 27

F

F9 Help key (telnet) 50
FDISK command (DOS) 18
features, PC-NFS 3–15
file access 10, 14, 68, 85
    transparent 9, 17
file attributes 19
    changing 26
    DOS 20, 25
    hidden, DOS 20
    mapping 19, 25–27
    NFS 21
    PC-NFS 26
file attributes, *continued*
    permissions in NFS 21
    read-only, DOS 20
    setuid 85
    user classes in NFS 21
file copy 113
file format 72, 76, 83, 120
    changing 14, 72
    defined 19
    DOS 14, 72, 120
    UNIX 14, 72, 120
file locking 30-33
    advisory 30
    mandatory 30
file management system 7
file names 20, 85
    mapping 27, 85
    remote 114
file ownership 25, 68
file protection 30
file servers
    defined 6
file sharing 3, 29–33, 101
file structure (ftp) 79
file system commands 14
file systems 7, 19
    defined 6
    displaying exported 15
    DOS 6
    exported 15, 101, 119
    joining 91
    mounting 10, 101
    mounting remote 15
    read-only 33, 101
    remote 7, 101
file transfer 3, 9, 18–19, 48, 74–83, 113
    in PC-NFS 12
    file transfer commands (ftp) 75–77
File Transfer Program, *see* ftp program
file, structure name (ftp) 80
files
    access 10, 14, 31, 68, 85
    advisory vs. mandatory locking 30
    ASCII 42, 76, 83
    attributes 19
    attributes mapping 19, 25–27
    attributes, DOS 20, 25
    attributes, NFS 21
    attributes, PC-NFS 26
    backing up and restoring 35–40
    changing attributes 26
    changing format 14, 72, 120
    changing permissions 14, 68
    conflicts 30
    converting 72, 120
    converting formats 14, 24, 24–25
    copying 113

- files, *continued*
  - copying between systems 18–19
  - copying to a remote printer 45
  - creation time 85
  - defined 6
  - Diablo-630 printer format 42
  - displaying information about 27, 85
  - DOS 6, 72, 83, 120
    - appending 75
    - on a remote system 76, 77
  - end-of-file, DOS 20
  - executable 14, 27, 68
  - extensions 27
  - format
    - conversion 14, 24–25, 72, 120
    - DOS 20
    - NFS 20–21
  - group permissions 68
  - hidden 85
  - locking 29
  - locking services, PC-NFS 29–33
  - mapping attributes 25–27
  - mapping DOS extensions 23
  - mapping names 22–24
  - modification time 85
  - moving 14
  - moving between directories 88
  - names in DOS 20
  - in a Network 17–27
  - NFS 19–21, 85
  - organization of 6, 19
  - ownership 14, 25, 68
  - permissions 14, 15, 68, 85
    - default 26
    - NFS 21
    - UNIX 25
  - PostScript format 42
  - printing 14, 96
  - printing on remote printers 42
  - printing UNIX files under DOS 44
  - remote 75
  - renaming 14, 24, 88
  - restoring an entire directory 40
  - restoring to NFS server 40
  - sharing 3, 29–33, 101
  - size 85
  - spooling 44, 96, 102
  - transparent access 9
  - UNIX 6, 72, 83, 120
  - user classes 68
  - user classes, NFS 21
- fonts, printer 42, 46
- form command (ftp) 79
- FORMAT command (DOS) 18
- formats, file 19-21, 24
- ftp program (PC-NFS) 12, 18, 19, 74–83
  - ! command 80
  - ? command, *see* help command
  - and SunOS 83
  - append command 75
  - ascii command 79
  - bell command 81
  - binary command 79
  - bye command, *see* quit command
  - binary representation 79, 80
  - cd command 77
  - close command 78, 83
  - commands 75–82
    - behavior 81–82
    - connection 78
    - file transfer 75–77
    - for your local PC 80–81
    - remote directory 77–78
    - transfer control 79–80
  - debug command 81
  - delete command 75
  - dir command 77
  - DOS commands from 80
  - executing DOS commands from 80
  - form command 79
  - get command, *see* recv command
  - glob command 81
  - globbing 74, 75
  - hash command 81
  - help command 81
  - lcd command 80
  - ls command 77
  - mdelete command 81, 83
  - mdir command 77, 83
  - mget command 77, 81, 83
  - mkdir command 77
  - mls command 78, 81, 83
  - mode command 79
  - mput command 77, 81, 83
  - open command 78, 83
  - port specification 74
  - prompt command 81
  - put command, *see* send command
  - pwd command 78
  - quit command 78
  - recv command 75
  - remotehelp command 82
  - rename command 76, 83
  - rmdir command 78
  - send command 76
  - sendport command 80
  - sessions 78
  - server 74, 78, 81, 82, 83

ftp program (PC-NFS), *continued*
- status command 82
- stream mode 79
- struct command 80
- take command 81
- tenex command 80
- type command 80
- user command 78, 83
- verbose command 82

ftpd daemon 83
function keys (telnet) 57-58

G

gateways 5, 14, 93, 97
get command (ftp), *see* recv command (ftp)
glob command (ftp) 81
globbing
- in ftp 74, 75

group (user class) 21
group id 69, 93
group name 26
groups
- group id 69

groups (user class) 68

H

hangup command (PC-NFS) 84
hard disks
- backing up files to local 39
- restoring files to local 40

hash command (ftp) 81
help
- from remote server (ftp) 82

help command (ftp) 81
Help key 50
hidden file attribute (DOS) 20
hidden files 85
hierarchical directories, *see* directories, tree structure
$HOME 102
home directory 102
host list
- telnet program 57

host name 5
Hot key
- printer 44
- telnet 49, 52, 53

I

I/O port address
- for controller boards 89

IBM PC-Network 14, 94
PC-NFS
- with IBM PC-Network 94

image representation in ftp, *see* binary representation in ftp
Internet address 14, 15, 66, 93, 95, 97, 98
- PC 93, 99

Internet Protocol 5, 47, 107
ICMP 107
IP, *see* Internet Protocol

J

JOIN command (DOS) 18
join, *see* net join command (PC-NFS)
joining network devices 14

L

LABEL command (DOS) 18
laser printers 42
lcd command (ftp) 80
LF, *see* line feed
LifeLine Backup 35
Light Emitting Diodes, *see* terminal emulation
line feed 20, 24, 72, 83, 120
LIST command (BASIC) 43
list directory contents 14
local
- defined 7
- disks 39–40
- printers 41

Local Mode, telnet program 60
local PC commands in ftp 80–81
lock manager 32
lock requests 30
locking files, *see* file locking
locking services 29–33
log in
- defined 5
- remote 3, 7, 13, 47, 78
- telnet 49
  - to a non-UNIX host 49
- to a network 14

log in name, *see* user name
log out, *see* net logout
logging off the network 14, 92
logical device names
- LPT1, LPT2, LPT3 10, 11, 43, 45, 96
- PRN 43, 102

Lotus 1-2-3 43, 44
lower case to upper case conversion 120
lp 44, 102
LPT1, LPT2, LPT3 10, 11, 43, 45, 96
ls command (ftp) 77
ls command (PC-NFS) 14, 24, 26, 85–87
lw 102

M

mandatory file locking 30
mapped file names 27
mapping DOS file name extensions 23
mapping file attributes 19, 25–27
mapping file names 22–24, 85
    examples 23
mask (subnet) 99
mdelete command (ftp) 77, 81, 83
mdir command (ftp) 77, 83
memory address, shared
    for controller boards 89
mget command (ftp) 77, 81, 83
MICOM-Interlan 89
Microsoft Windows 103
mkdir command (ftp) 77
mls command (ftp) 78, 81, 83
mode command (ftp) 79
modems 71, 84
modes
    compatibility 31
    Diablo-630 printer 46
    Read Only 33
    sharing 31, 32
    spooling 44
modification time (files) 85
mount command (UNIX) 18, 101
mounting
    defined 10
    file systems 10
    printers 10
    remote file systems 101
mounting file systems 15
moving between operating systems 17–27
moving files 14, 88
mput command (ftp) 77, 81, 83
MS-Net support 29
/MS switch, *see* /MUSTSHARE switch
multiuser environment 20, 30
MUSTSHARE 32, 33
mustshare mode, file locking 32
/MUSTSHARE switch 101
mv command (PC-NFS) 14, 24, 88
myeaddr command (PC-NFS) 89

N

net blip command (PC-NFS) 14, 90
net join command (PC-NFS) 18, 91
net logout command (PC-NFS) 14, 92
net name command (PC-NFS) 14, 26, 92, 93, 95, 102
net pcnet command (PC-NFS) 14, 94
net pcnfsd command (PC-NFS) 14, 93, 95
net print command (PC-NFS) 14, 42, 43, 45, 96, 102
net route command (PC-NFS) 14, 97
net start rdr command (PC-NFS) 14, 98, 99
net stop rdr command (PC-NFS) 14, 98
net subnet command (PC-NFS) 14, 99
net umask command (PC-NFS) 15, 27, 100
net use command (PC-NFS) 10, 15, 30, 32, 43, 46, 91, 95, 101–103
    printer commands 95
net version command (PC-NFS) 15, 104
net ypdomain command (PC-NFS) 15, 105
net ypset command (PC-NFS) 15, 93, 95, 106
NETBIOS support 29
netstat command (PC-NFS) 15, 107
network
    activity indicator (blip) 90
    administrator, *see* system administrator
    ASCII 79, 80
    commands 15, 90–106
    compatibility 94
    connections (telnet) 56
    date 93
    defined 4
    drives, *see* remote file systems
    environment controls 14
    files 17–27
    illustrations 5, 10, 11
    interface 15
    log out 92
    manager, *see* system administrator
    overview 4–7
    printers, *see* remote printers
    statistics
    time 93
    troubleshooting 15
    usage statistics 110
new line, *see* line feed
NFS 15
    authentication 93
    authentication server 95
    file names 85
    file protection 68
    files 19–21, 85

NFS, *continued*
- permissions 100
- server 15, 108, 116
- statistics 15, 110
- usage 110

NFS server
- backing up files from 36
- backing up files to 37–39
- maintaining a backup directory on 38
- print services available on 41
- restoring files to 40
- storing DOS programs on 27

nfsconf, *see* configuration program
\NFS\HOSTS file 98
\NFS\NETWORK.BAT file 93, 95, 98, 105, 106
nfsping command (PC-NFS) 15, 108–109
nfsstat command (PC-NFS) 15, 110–112
NI5010 89
NIC 89
NIU 89
No Echo Mode, telnet program 60
nobody user name 26, 92
noname domain name 105
non-print format (ftp) 79

O

open command (ftp) 78, 83
operating systems 19
operating systems environments 17–27
others (user class) 21
owner (file) 68
ownership (file) 25

P

parallel printing devices, *see* LPT1, LPT2, LPT3
password file 102
path names 103
- defined 6
- DOS 20
- illustration 7
- PC printers 41

PC applications
- escaping to DOS 44
- record locking 31

PC commands in ftp 80–81
PC name 14
PC-Network (IBM) 14, 94
PC-NFS
- and DOS 4
- and DOS file sharing modes 31
- client-only 6

PC-NFS, *continued*
- commands 14-15, 65–123
- compatibility 94
- concurrency 14
- defined 4
- overview 3–4
- printer services 41–46
- starting 14, 98
- stopping 14, 98
- version number 15, 104

PC-NFS features 3–15
pcnfsd daemon 14, 93, 95
permissions 15, 30, 85
- changing default 27
- changing for file 27
- default 26, 100
- default, changing 27
- files, UNIX 25
- NFS 100
- for NFS directories 27
- NFS files 21
- sticky 69

phone lines, *see* serial connections, dial-up
pica font 42
ping, *see* nfsping command
pipes under DOS 27
port specification
- for ftp 74

PostScript print files 42, 46
PRINT command (DOS) 43
print command, *see* net print
print servers 6
printers
- buffer 42
- commands 95
- copying files to 45
- Diablo-630 format 42
- DOS 10
- fonts 42
- Hot key 44
- laser 42
- local 41
- LPT1, LPT2, LPT3 10
- network, *see* remote printers
- PostScript format 42
- ps630 emulation 46
- redirecting 43, 45, 96, 101
- remote 7, 9, 10, 11, 42, 44, 96, 101
- services 41
- spool 44
- Sun LaserWriter 42, 46
- time-outs 44, 46

printing 3, 14, 41-46
- ASCII files 42
- Diablo-630 format files 42
- files 96
- from applications 11

printing, *continued*
  on remote printers 42
  PostScript format files 42
  pstext mode files 46
  redirected 43, 45, 96, 101
  UNIX files under DOS 44
  with PC-NFS 41–46
  with rsh 118
PRN 43, 102
programs
  EDLIN 27
  ftp 12, 18, 19, 74–83
  remote copy program, *see* rcp command (PC-NFS)
  storing on NFS server 27
  telnet 13, 47–63
prompt command (ftp) 81
protection against unauthorized users 27
protocols 5, 47
  ARP 5, 66, 107
  Ethernet 4
  Internet 5, 47, 107
  ICMP 107
  network 107
  NFS 110
  RARP 98
  RPC 110, 116
  statistics 15
  TCP 19, 107
  TCP/IP 47
  UDP 107
PrtSc (Print Screen) key 43
PS/2 89
ps630 emulation software 46
pstext mode 46
put command (ftp), *see* send command (ftp)
pwd command (ftp) 78

Q

quit command (ftp) 78

R

RARP 98
rcp command (PC-NFS) 12, 18, 19, 113–115
  and wildcards 114
  file name format 113
read access 31
read permission 21, 69
read-only file attribute (DOS) 20
read-only file systems 33, 101
read-only permission 68
/READONLY switch 33, 101
record locking 30, 31
RECOVER command (DOS) 18
recv command (ftp) 75
redirecting printing 43, 45, 96, 101
remote
  access 12, 13, 47–63, 74-83, 117
  defined 7
  directories 78
  file access 10, 12, 74–83
  file names 114
  file systems 7, 101
    joining 91
    mounting 15, 101
  file transfer 113
  files 75, 77
    deleting 75, 77
  help 82
  log in 3, 7, 13, 47, 78
  printers 7, 9, 10, 11, 42, 44, 96, 101
  shell 48
  system access 9, 13, 117
remote command execution 13
remote copy program, *see* rcp command
remote directory commands (ftp) 77–78
remote procedure calls, *see* RPC
remote shell 117
remote help command (ftp) 82
remote system access 47–63
RENAME command (DOS) 24
rename command (ftp) 76, 83
renaming files and directories 14, 24, 76, 83. 88
representation type (ftp) 79
resource sharing 4
RESTORE command (DOS) 37, 40
restoring files 35–40
  to local hard disk 40
  to NFS server 40
Reverse ARP 98
rmdir command (ftp) 78
/RO switch, *see* /READONLY switch
RPC 15, 110, 116
  usage 110
rpcinfo command (PC-NFS) 15, 116
RS232, *see* serial connections
rsh command (PC-NFS) 13, 48, 117–118
  printing 118
rshd daemon 12, 13

S

send command (ftp) 76
sendport command (ftp) 80

- serial connections 3, 4
  - dial-up 71, 84
  - `telnet` 48, 56, 61
- servers
  - authentication 14, 15, 93, 95
  - defined 6
  - exported file systems 15
  - file 6
  - `ftp` 74, 78, 81, 82, 83
  - NFS 15, 108, 116
  - print 6
  - `telnet` 47
  - VMS 103
  - Yellow Pages 14, 15, 93, 102, 106, 116
- sessions 19
  - `ftp` 78
  - `telnet` 48–53
- `SET` command (DOS) 62
- `setdomain` command (SunOS), *see* `net ypdomain` command (PC-NFS)
- `setenv` command (UNIX) 50
- `setuid` attribute (UNIX) 85
- `SHARE` 32
- `/SHARE` switch 101
- shared data bases 29
- shared memory address
  - for controller boards 89
- shared resources 4
- sharing mode 32
- sharing file systems 101
- sharing files, *see* file sharing
- sharing mode, file locking 32
- shell, remote 48
- `showmnt` command (PC-NFS) 15, 119
- `/SH` switch, *see* `/SHARE` switch
- single-user environment 20
- size (file) 85
- slashes 103
- snow on PC screen 60
- spooling 44, 96, 102
- SSI WordPerfect 43, 44
- starting PC-NFS 14, 98
- statistics
  - network 15, 107
  - network protocols 15
  - network usage 110
  - NFS 110
  - NFS protocol 15
- status command (`ftp`) 82
- sticky permission bit 69
- stopping PC-NFS 14, 98
- storing programs on NFS server 27
- stream mode (`ftp`) 79
- `struct` command (`ftp`) 80
- subnet mask 14, 99
- `SUBST` command (DOS) 18
- Sun LaserWriter 42
- superuser 103
- `SYS` command (DOS) 18
- system access
  - remote 9, 13, 117
- system administrator 5, 26, 35, 41, 101

T

- tabs, setting for `telnet` 61
- take command (`ftp`) 81
- TCP and TCP/IP 19, 47, 107
- escaping to DOS
  - from `telnet` 53
- `telnet` program 13, 47–63
  - after a crash 62
  - beginning a session 48
  - Change Parameters Menu 60
  - changing connections 52
  - changing environments 53
  - com ports 48
  - command syntax 48
  - connecting to a host system 61
  - Connection Type Menu 56
  - control menus 55, 57
  - daemon 13, 48
  - deactivating 53
  - displaying host system name 61
  - Echo Mode 60
  - `em.ses` file 62
  - ending a session 52
  - Exit Menu 52, 55
  - File Menu 58
  - function keys 57-58
  - Help key 50, 58
  - Hot key 49, 52, 53
  - initialization menus 55
  - local mode 60
  - log in 49
    - to a non-UNIX host 49
  - logging in to a host system 49
  - mailing a DOS file 63
  - menus 55–61
  - network connections 56
  - No Echo Mode 60
  - problems 63
  - receiving files 59
  - recording session output 58
  - removing from memory 53
  - resetting VT100 characteristics 60
  - resident in memory 52
  - returning to DOS 53
  - Select a Host Menu 56
  - selecting terminal mode 60
  - Send File options 58

telnet program, *continued*
sending files between environments 58
serial connections 48, 56, 61
server 47
sessions 48–53
setting auto-line feed 60
setting tabs 61
specifying terminal types 50
starting from a batch file 49
storing previous settings 62
Telnet Command Menu 61
Telnet Host List 57
Telnet Status Line 49, 60
Terminal Emulation Menu 56
Toggle Carriage Return Mode 61
Video Retrace Wait 60
telnetd daemon 13, 48
TENEX 79, 80
tenex command (ftp) 80
terminal
emulation 47, 49, 51
resetting characteristics 60
text files, converting 24
time (network) 93
time-outs, printer 44, 46
time zone (network) 93
TNHOSTS file (telnet) 57
TRANSCRIPT, Adobe Systems 45
transfer control commands (ftp) 79–80
transfer mode (ftp) 79
transferring files, *see* file transfer
Transmission Control Protocol, *see* TCP and TCP/IP
transparent file access 9, 17
tree structure, directory 6
troubleshooting 15
type command (ftp) 80
typewriter font 42

U

unauthorized users
protection against 27
Ungermann-Bass 89
UDP 107
UNIX
Berkeley 4.2 BSD 3
df command 103
file format 72, 76, 83, 120
file systems 6
files 6, 83, 85
mount command 101
printing files under DOS 44
required for rcp 12
vs. DOS file formats 24

unix2dos command (PC-NFS) 14, 25, 44, 120–121
upper case to lower case conversion 72, 120
user classes 30, 68
NFS files 21
user command (ftp) 78, 83
user id 93
user name 26, 93
default 26
defined 5
nobody 26, 92

V

verbose command (ftp) 82
version number
PC-NFS 15, 104
PC-NFS commands 65
VMS 20
servers 103
-v option switch (PC-NFS commands) 65
VT100 emulation 13
VT100 terminal 48, 50
emulation 51

W

WD8003E 89
Western Digital 89
wildcards 45
in ftp 83
in rcp 114
ls command 85
Windows, Microsoft 103
WordPerfect, SSI 43, 44
Wordstar, Micropro 120
working directory 77, 78, 80
write permission 21, 69

Y

Yellow Pages 98
database 119, 122, 123
defined 5
displaying values 15
domain 15, 93, 105
searching 15
server 93, 102, 106, 116
setting domain 15
setting server 15
Yellow Pages server 14
YP, *see* Yellow Pages
ypcat command (PC-NFS) 15, 122
ypmatch command (PC-NFS) 15, 123
$YPSERVER 102

To return this card, cut along the dotted line.

Fold so that the address on the other side appears.

Place a first class stamp in the space provided.

PC-NFS User's Manual Comment Form

Please return this card and help us improve our manuals. Your comments and suggestions will be incorporated into subsequent editions of this manual.

Did you find this manual:

	Yes	No
Well organized	□	□
Complete	□	□
Easy to Use	□	□
Technically Accurate	□	□
Well Written	□	□
Error Free	□	□
With Adequate Examples	□	□

Comments____

____

____

____

____

____

Name____

Company____

Address____

____

____

Sun Microsystems
Two Federal Street
Billerica, MA 01821
Attn: Information Architecture Group

PC-NFS 3.0.1
Change Pages

PC-NFS User's Manual

Part No. 814-1031-02

Instructions

1. Take the Read This First document out of your documentation binder and replace it with the 3.0.1 Release Notes.

   If you are upgrading from 1.0 or 2.0, your new documentation does not include a copy of the Read This First document. In this case, place the 3.0.1 Release Notes in the front of the binder.

2. Use the following instructions to replace pages in PC-NFS User's Manual.

Take out pages	Insert pages
iii-iv	iii-iv
1-4	1-4
45-46	45-46
59-62	59-62
69-70	69-70
77-78	77-78
85-86	85-86
89-90	89-90
93-94	93-94
101-104	101-104
117-118	117-118

You may want to file this instruction sheet at the back of your binder to keep a record of the changes.

sun
microsystems

Contents

Introduction 1

Chapter 1 – PC-NFS Overview 3

DOS Users and PC-NFS 4

Networks and Network Terminology 4

Servers 6

Directories and File Systems 6

Local and Remote 7

Summary 7

Chapter 2 – PC-NFS Features 9

Transparent File Access 9

File Transfer 12

The File Transfer Program – `ftp` 12

The Remote Copy Program – `rcp` 12

Remote System Access 13

The `rsh` Command 13

The `telnet` Facility 13

File System Commands 14

Network Management 14

Network Statistics and Troubleshooting Commands 15

Chapter 3 – Files in a Networked Environment **17**
DOS Commands and Networked Disks 18
Copying Files Between Systems 18
Accessing Files from Different Operating Systems 19
  DOS Files 20
  NFS Files 20
Bridging the Systems 22
  Mapping File Names 22
  Converting Text File Formats 24
  Mapping File Attributes 25
  Running DOS Programs on an NFS Disk 27

Chapter 4 – File Locking **29**
Who Should Read This Chapter 29
PC-NFS File Locking Services 30
  Advisory File Locking 30
  File Sharing 31
  Record Locking 31
Using PC-NFS Locking Services 32
  Read-Only File Systems 33

Chapter 5 – Backing Up and Restoring Files **35**
Choosing a Backup Method 35
Backing Up Files to an NFS Server 37
Backing Up Files to Your Local Hard Disk 39
Restoring Files to Your Local Hard Disk 40
Restoring Files to an NFS Server 40

Chapter 6 – Using Printers **41**
Using Your Local Printer 41
Using Remote Printers 42
  Determining Your Printing Options 42
Redirecting DOS Printer Output 43
  If You Don't Know the Name of a Remote Printer 44
  Redirected Output 44
  Printing with `net print` 45
  Using a Sun LaserWriter 45

Introduction

PC-NFS is a set of applications designed to help you get the most out of your PC and network environment.

This book is designed to introduce PC-NFS concepts and features, and includes a complete command reference, glossary and index. For information about installing and configuring PC-NFS as well as a troubleshooting guide and technical reference, see *Installing PC-NFS, A Guide to the User and System Administrator.*

This book, the *PC-NFS User's Manual*, introduces networks and terms in Chapter 1, and provides an overview of PC-NFS features in Chapter 2. Chapter 3 discusses files in a networked environment and Chapter 4 describes file locking under PC-NFS. Chapter 5 describes backing up and restoring your files across the network. Chapter 6 tells you about printing files from PC-NFS. Chapter 7 describes the network applications, `telnet` and `rsh`. Chapter 8 is the Commands Reference; it lists all PC-NFS commands in alphabetical order.

Also available are PC-NFS LifeLine, electronic mail and backup for use with PC-NFS, and The PC-NFS Programmer's Toolkit. For more information about these products, contact your Sun representative.

This manual uses the following conventions:

Italic print	to introduce a new term, or for emphasis
`Courier`	to indicate command or program names or output on a screen
`Bold Courier`	to indicate what you, the user, type

PC-NFS Overview

1

This chapter introduces the personal computer version of Sun Microsystems's Network File System (PC-NFS). PC-NFS is a PC networking product that allows you to:

- Share your files with other DOS users on your network, without exchanging diskettes.
- Transfer files between systems using commands such as the DOS `COPY` command, or the PC-NFS commands `rcp` or `ftp`.
- Share files with users of different operating systems in your NFS network, including UNIX operating systems and VMS.
- Use the file sharing and locking services provided by DOS 3.1 (but not by the products NETBIOS or MS-Net).
- Print files on the network's printers.
- Remotely log into non-NFS systems using `ftp`, `rsh`, and `telnet`.
- Use selected networking commands compatible with the Berkeley 4.2BSD UNIX operating system.
- Access all of these facilities either directly on a network or over a serial line, such as a phone line or RS232 line.

If you are familiar with networks and their terminology, you may want to skip the rest of this chapter and proceed to Chapter 2, *PC-NFS Features*. The remainder of this chapter discusses PC-NFS from a DOS user's point of view and introduces the concept of a network and its associated terminology.

DOS Users and PC-NFS

As a DOS user, the usual way to transfer a file to another DOS user on another computer system is to copy the file to a diskette and hand-carry the diskette to the other system. If you could link the two systems together you could not only make the file transfer easier, you could do more than transfer files.

You could:

- Read directories and files that reside on another non-DOS computer
- Run applications on your computer that reside on another computer
- Use a laser printer connected to another computer
- Backup files and directories to shared resources, such as magnetic tape drives
- Access databases that are too large for one PC-based system

PC-NFS is a set of applications and commands that give you the ability to do all of these tasks. Moreover, it gives the ability to connect to machines that do not use DOS.

To understand how PC-NFS accomplishes these tasks, you must be familiar with networks. The next section provides a brief introduction to networks.

Networks and Network Terminology

A *network* is a mechanism that links together two or more computers to share resources. The most commonly shared resources are:

- information, in the form of files
- printers, such as a laser printer
- processing power, such as running a program on another machine

A network links together the various hardware devices (computers, printers, and backup devices) you want to share. It requires both a physical connection between the linked devices and software. The physical link is a line, not unlike a telephone line, and the software is some type of network software package.

There are different types of computer networks differentiated at the basic level by the type of wire and electronics used to connect the computers. At this level, PC-NFS works with *Ethernet* networks. Instead of Ethernet, you can use the serial port on your PC to connect your PC to a machine on a network that supports NFS. The network can be just your machine and the machine running NFS.

Printing with `net print`

Once you have redirected your local printer's output, you print a text file on a remote printer by *copying* it. For example, to print a file called `yourfile`, you enter:

```
C>COPY yourfile LPT1
```

This is a somewhat slow process for a large file, however. For more time-efficient print operations, use the PC-NFS `net print` command:

```
C>net print yourfile
```

If you don't specify a print device, PC-NFS assumes you want your file to print on `LPT1`. You can send your file to `LPT2` or `LPT3` by including `LPT`*n*: at the end of the preceding command line (you need the colon).

You can issue multiple file name arguments in the `net print` command, and you can use DOS wildcard characters (`*` and `?`) in the file name arguments. When you print multiple files, your output is a single print job with form feeds separating the files.

Note that when the `*` character is used alone as a `net print` argument, the `*` represents spooled files, not a wildcard for all files. See the section, *Using Remote Printers*, earlier in this chapter for more information.

Remember: The `net print` command works with redirected print devices only. If you have not redirected the output for printer `LPT`*n*:, you'll see the following NFS system message:

```
NFS041F : You have not done a net use LPTn: \\host\printername.
```

Using a Sun LaserWriter

If you have configured a Sun LaserWriter, the following sections may be of interest to you.

PC-NFS printing is optimized for use with the Sun LaserWriter, available in a package that includes UNIX-compatible TRANSCRIPT™ software from Adobe Systems.

PC-NFS supports three ways of printing on the laser writer printer:

- As a line printer emulator
- As a PostScript printer
- As a Diablo-630 printer emulator

Sun LaserWriter as a Line Printer. When you simply redirect printer output with the `net use` command, the Sun LaserWriter prints your data in *pstext* mode using a simplified fixed-width font. This format is the same one you get when you copy a file to an NFS server and print it with `lpr(1)`.

Sun LaserWriter as a PostScript Printer. If you have a PC application that can generate PostScript commands, you can use a remote Sun LaserWriter with that application.

Sun LaserWriter as Diablo-630. You can use Diablo-630 printer emulation on a Sun LaserWriter with applications such as word processing packages that can take advantage of a high-quality printer with proportional fonts and micropositioning. To use Diablo-630 emulation, first configure your application for this type of printer. Then use the PC-NFS configuration program, `nfsconf`, to select `Diablo 630` mode and configure the emulated printer.

Note that the Diablo-630 emulation software (**ps630**) does not currently support the following features:

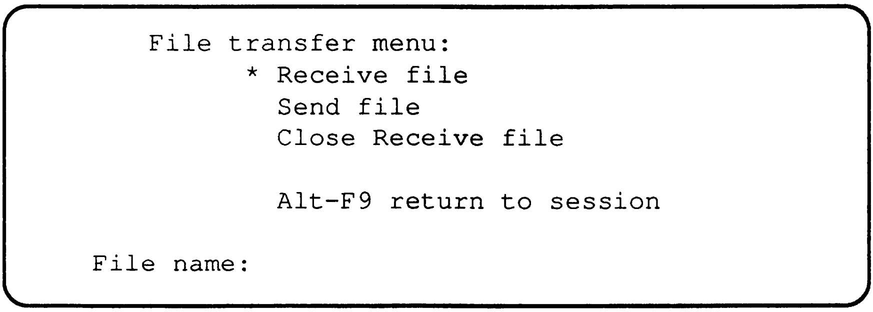
- Print suppression
- HY-plot
- Extended character set
- Downloading print wheel information or program mode
- Page lengths other than 11 inches
- Paper feeder control
- Hammer energy control
- Remote diagnostics
- Backward printing control

ps630 software *does* support reverse printing and bold output produced with the correct (Ctrl-O) escape sequence. Note that double-striking doesn't produce bold output. For more information about this software, see the manual page for `ps630(1)` on the NFS server.

- Close Receive file – stops recording session output in the previously created Receive file.

To display the File Menu, press the Alt and F9 keys simultaneously.

The File Menu allows you to transfer information between your PC and your telnet host. The File Menu looks like this:



```
          File transfer menu:
                * Receive file
                  Send file
                  Close Receive file

                  Alt-F9 return to session

      File name:
```

Select an option by positioning the asterisk next to your selection using the up arrow and down arrow keys.

To send a file to your telnet host:

1. Begin your telnet session.
2. On the remote system, prepare to receive a file. For example, create a new file using the cat command: cat > destinationfile
3. Press the Alt and F9 keys simultaneously.
4. Select Send file.
5. Enter the file name.
6. Close the receiving file on the remote system. For example, if you were receiving the file using cat, use a Ctrl D to close the file.

To receive a file from your telnet host:

1. On the remote system, prepare to send a file to stdout. For example, cat < sourcefile. Do *not* press the Enter key.
2. Press the Alt and F9 keys simultaneously.
3. Select Receive file.
4. Enter the file name.
5. On the remote system, press the Enter key.
6. When you are ready to close the file, press the Alt and F9 keys simultaneously.
7. Select Close Receive file.

Note that as long as your receive file is open, the status line at the bottom of your telnet screen displays the name of your receive file in your status line in the format >filename.

You may prefer to use the ftp command to send and receive files between your PC and remote systems. See Chapter 8 for a discussion of ftp.

Change Parameters Menu

The Change Parameters Menu lets you control the following terminal display characteristics:

- Terminal Mode – offers you three choices: No Echo, Echo, and Local.

  Generally, you should select No Echo mode. This is because most host systems automatically echo the characters you enter on your PC screen. However, some hosts do require you to select Echo mode. Ask your system administrator which option is appropriate for you.

  Local mode prevents telnet from sending the characters you enter to the host system. You can reset your terminal to Local mode at any time by pressing the Alt and F6 keys simultaneously.

- Status Line – controls whether or not the bottom line of your screen displays the telnet Status Line. The section *Logging in to a Host System* describes the information that this status line displays.

- Auto Line Feed – causes remote hosts that don't echo line feeds to your terminal to do so. If you are connecting to a non-UNIX host, you might need to set this option.

- Video Retrace Wait – lets PCs requiring Video Retrace Wait to function correctly. If your PC requires Video Retrace Wait and you *don't* set this option, your screen displays "snow."

  When Video Retrace is set to "on," you might notice a slower response from your PC.

- Xon/Xoff – enables and disables the use of Ctrl-S and Ctrl-Q for controlling the flow of your screen display. The default is Xon.

- Reset Terminal Setup – allows you to reset to standard VT100 terminal characteristics. This is similar to the Reset key on the VT100 keyboard. This feature is most useful when the terminal emulator is in an unusual state due to binary data sent to it.

- Tabs setting – lets you set tabs on your screen. Use the Tab key to move the cursor to the bottom of your screen, where the tab settings are displayed. Then, use the arrow keys to move from column to column, and the Enter key to select the desired tab settings. You can also use the Enter key to cancel existing tab settings.

To display the Change Parameters Menu, press the F6 key.

Serial Port Use

If you are using a serial port on your PC to connect to a host system, the Change Parameters Menu displays two additional items:

- Baud Rate – sets the speed at which data moves across the communications line.
- Parity – sets the type of error checking that the communications line uses.

You must configure your PC's serial port so that it matches your connection with the host system. If you aren't sure how to set these parameters, check with your system administrator.

The telnet Command Menu

This menu is available only when using telnet over the Ethernet.

The Telnet Command Menu provides the following telnet control functions:

- Re-open Connection – reconnects your PC to the host system. Select this item if your connection to the host system was broken by an unusual event, such as a problem on the network.
- Display Status – displays the name of the host system to which you are connected.
- Toggle Carriage Return Mode – causes your terminal to perform a carriage return whenever you press the Enter key.
- Exit Command Menu – returns you to your session on the host system.

To display the Telnet Command Menu, press the F8 key.

Storing Previous `telnet` Settings

The first time you use `telnet`, the program creates a file called `em.ses`. This file contains the `telnet` settings you select from the initialization and control menus (for example, the baud rate).

When you begin subsequent sessions, `telnet`:

- Uses the contents of the `em.ses` file as default settings for your current session.
- Replaces existing `em.ses` settings with any new options you select during your current session.

The `em.ses` file thus spares you the need to select the same menu options at the beginning of every `telnet` session.

If your `telnet` host crashes or your `telnet` session has an abnormal termination, delete your `em.ses` file before you restart `telnet`. `telnet` will create a new `em.ses`.

The `em.ses` File Directory. By default, `telnet` stores the `em.ses` file on the drive where PC-NFS is installed in the directory `\NFS`.

If you want to store `em.ses` elsewhere, use the following command line from DOS:

```
C>SET em.ses=path
```

where *path* is the location of the directory in which you want `em.ses` to reside.

If your PC doesn't have a directory called `\NFS` and you don't specify a location for the `em.ses` file, `telnet` creates `em.ses` in your current directory.

> **Never edit your `em.ses` file directly. Let `telnet` create and maintain this file.**

You don't need to take any active steps to keep your `em.ses` file up-to-date; `telnet` does it all for you. And if you accidentally delete your `em.ses` file, `telnet` creates a new file at the beginning of your next session.

= assign assign *exactly* the specified permissions to the class. Those permissions that are specified are added, and any not specified are removed.

3. The *permissions* [`rwxstugo`] that are changed or assigned:

`r` read permission to read the contents of the file

`w` write permission to write to the file

`x` execute permission to execute a file if it is executable or a UNIX shell script, or if the file is a directory, permission to search in the directory.

`s` set id set the owner or group id. This is useful only if the user is the owner (`u`) or in the owner's group (`g`) and the file is on a system running under UNIX.

`t` sticky The sticky permission enables you to speed up the execution of a file that is frequently used by causing the system to keep the file in memory or the swap area between executions.

`u`, `g`, or `o` The letters `u`, `g`, or `o` indicate that the permissions are to be taken from the current mode for that user class.

The symbolic mode may also be made up of several of these pieces, separated by commas. For example, to give yourself read, write, and execute permission, and all others only read permission, you could use `u=rwe,go=r`.

Absolute Mode. Instead of using letters to designate who is able to access the file or execute it, you can use the *absolute mode*. The absolute mode is an octal number, made up of values as follows:

4000	set user id on execution
2000	set group id on execution
1000	sticky bit
0400	read by owner (`u=r`)
0200	write by owner (`u=w`)
0100	execute (search in directory) by owner (`u=x`)
0700	read, write, execute (search) by owner (`u=rwx`)
0070	read, write, execute (search) by group (`g=rwx`)
0007	read, write, execute (search) by others (`o=rwx`)

Examples

To make a shell script called humbug executable using a symbolic mode, enter:

```
D>chmod +x humbug
```

The command example above changes the modes (chmod) of the file by making it executable (+x).

To see what permissions are set on file, enter ls -l

```
C>ls -l
total 194
-rwxrwxrwx 1 tupjohn   28434 Aug 12 12:52 humbug
drwxrwxrwx 2 tupjohn     512 Aug 14 14:16 ORIGINALS
-rw-rw-rw- 1 tupjohn      70 Aug 12 13:02 form
```

The access permissions are represented by the letters in the first column. They appear in groups of three letters in the order of: owner's permissions, group permissions, and others' permissions. A hyphen (–) before a letter indicates that the mode is not permitted.

To make the shell script, humbug, executable by owner, group, and others using the absolute mode (equivalent to a=rwx, or ugo=rwx), enter:

```
D>chmod 0777 humbug
```

See Also

ls command, later in this chapter.

net umask command, later in this chapter.

UNIX chmod(1) documentation.

Discussion of NFS file attributes in Chapter 3.

`mdelete` *remote-files ...*

Delete the *remote-files* on the remote machine.

`mget` *remote-files ...*

Expand wildcards in the list of *remote-files...* on the remote machine (if globbing is turned on), and do a `get` for each file name thus produced. Files are transferred into the local working directory.

`mput` *dos-files ...*

Expand wildcards in the list of *dos-files* (if globbing is turned on), and do a `put` for each file name thus produced.

Remote Directory Commands. These commands enable you:

- to establish your working directory on the remote machine
- to display file names on the remote machine
- to create and to delete directories (if the permissions allow that).

Note that remote file names and remote directories must conform to the remote host's operating system format when using these commands.

`cd` *remote-directory*

Change the working directory on the remote machine to *remote-directory.*

`dir` [*remote-directory* [*dos-file*]]

Produce a listing of the directory contents for the directory *remote-directory* on the remote machine, and, optionally, place the output in *dos-file.* If no *remote-directory* is specified, the current working directory on the remote machine is used. If no *dos-file* is specified, or if it is –, the listing is displayed on the terminal.

`ls` [*remote-directory* [*dos-file*]]

Produce an abbreviated listing of the directory contents for the directory *remote-directory* on the remote machine, and, optionally, place the output in *dos-file.* If no *remote-directory* is specified, the current working directory on the remote machine is used. If no *dos-file* is specified, or if it is –, the listing is displayed on the terminal.

`mdir` [*remote-files* [*dos-file*]]

Like `dir`, except multiple *remote-files* can be specified, and if globbing is turned on, wildcards can be used.

`mkdir` *remote-directory-name*

Create a directory named *remote-directory-name* on the remote machine.

`mls` [*remote-files* [*dos-file*]]

Like `ls`, except multiple *remote-files* can be specified, and if globbing is turned on, wildcards can be used.

`pwd`

Display the name of the current working directory on the remote machine.

`rmdir` *remote-directory*

Delete the directory *remote-directory* on the remote machine. If this cannot be done for some reason, such as the directory still contains some files, an error message displays.

Connection Commands. These commands allow you to establish and terminate connections to a remote host.

`bye`

Same as `quit` command, see the `quit` command.

`quit`

Terminate the ftp session with the remote server by disconnecting from it (like the `close` command), and exit `ftp`.

`close`

Terminate the ftp session with the remote server by disconnecting from it, and return to `ftp`'s command interpreter for another command. This allows you to start another session with another host.

`open` *host* [*port*]

Establish a connection to the specified *host*'s `ftp` server. You can supply an optional *port* number, in which case, `ftp` attempts to contact an `ftp` server using that port on the *host.*

If you specify a *host* name on the command line when you start `ftp`, it automatically tries to connect to that *host* as if your first command were `open` *host.*

`user` [*user-name* [*password* [*account*]]]

Identify yourself to the remote `ftp` server. If you do not specify a *user-name*, `ftp` prompts you for it, with the default being your current PC-NFS log in name (from the `net name` command). If you do not specify a *password* and the server requires it, `ftp` prompts you for it (after disabling local echo). If you do not specify an *account*, and the `ftp` server requires it, the `ftp` prompts you for it.

`ls` Display File Information

The `ls` command displays information about files. It displays the name, size, creation date, modification time, and access permissions for files.

Command Format

`ls` [`-a`] [`-d`] [`-b` | `-l` | `-u` | `-w`] [*filename* | *directoryname*]...

where:

filename, if present, is a DOS file name or wildcard pattern that selects which files to display information about.

directoryname, if present, is a DOS directory name. The `ls` command displays the file information for all files in that directory (except with the `-d` switch, see below).

Options

Note that the `-b`, `-l`, `-u` and `-w` options are mutually exclusive.

With no options specified, the `ls` command produces a file listing similar to the DOS `DIR` command, showing for each file the DOS name, size, and modification date and time. In addition, it shows the UNIX-style owner permissions, and the NFS file name if the DOS name is a *mapped name*.

`-a` Include hidden files in the listing — those with the hidden bit set in DOS, or for which the UNIX `setuid` attribute is set. The `-a` option can be combined with any of the others, e.g. `-al`.

`-d` Display directories as files. Normally `ls` displays the *contents* of a directory when the *directoryname* appears on the command line. If you use the `-d` option, it lists the information about the directory itself. You can combine the `-d` option combine with any of the others, e.g. `-dl`.

`-b` For each file, list the DOS name and NFS name, side by side. This can be used by a program that wants to operate upon both types of names.

`-l` Produce a listing very similar in form to the UNIX `ls -l` command, showing for each file the NFS file name, size, protection, modification date, and ownership, one file per line.

`-u` (Unix) Same as the `-l` option.

`-w` Produce a listing that is similar to the DOS `DIR/W` command, which is just the DOS file names in columns, without the volume and directory name.

Examples

ls command entered without options.

```
        1      2      3        4         5       6       7
```

```
C>ls
.        <DIR>              1-01-80  12:04a  U-rwx
..       <DIR>              1-01-80  12:04a  U-rwx
DOS2UNI   EXE     10086     4-15-86   3:48p  U-rwx
EM        COM     34398     4-16-86  11:48a  U-rwx
3C501     300      3459     4-15-86   3:24p  U-rwx
NATURE               56     4-15-86   2:58p  U-rwx
RICHES               48     4-15-86   2:58p  O-rwx
SPEED~CA            398     1-09-86   2:35p  U-rwx  speed.dial
THOUGHT              75     4-15-86   2:58p  U-rwx
EM        SES       574     1-01-86   2:40a  U-rwx
```

Column	Description
1	DOS file or directory name
2	Extension
3	Size in bytes
4	Creation or modification date
5	Creation or modification time
6	Permissions
7	NFS name, if different

The –b style produces a listing with two columns. The first column is the filename in DOS format; the second column is the filename in UNIX format. This illustrates the name mapping between DOS and UNIX names.

```
C>ls -b
..            ..
DOS2UNIX.EXE  dos2unix.exe
3C501.300     3c501.300
NATURE        nature
RICHES        riches
SPEED~CA      speed.dial
THOUGHT       thought
EM.SES        em.ses
```

myeaddr — Display Ethernet Address

This command displays the Ethernet address of your PC.

Command Format

myeaddr *controller_type* [*ioport_addr* | *shmem_addr* [*interrupt#*]]

where:

controller_type is the option name of the communications controller on your system, either the **3C501, 3C503, 3C505, 3C523, NIC, NIU, NI5010**, or **WD8003E.**

ioport_addr is the I/O port address for 3C501, 3C503, 3C505 or NI5010 controllers. This argument should be used only when the factory settings have been changed from the default settings.

shmem_addr is the shared memory address for NIC controller boards. This argument should be used only when the factory settings have been changed.

interrupt# is the interrupt number for the controller boards. This argument should be used only when the factory settings have been changed.

Examples

```
C>myeaddr 3c501
Your 3c501 Ethernet address is: 2:60:8c:15:90:96

C>
```

See Also

See the Installation section of *Installing PC-NFS, A Guide to the User and System Administrator* for a discussion of how to determine the *ioport_addr*, *shmem_addr*, or *interrupt#* addresses.

`net blip` Turn Blip On or Off

The `net blip` command turns the network activity indicator on or off. When it is on, a rectangle (blip) is displayed in the upper right corner of the screen at the start of every NFS remote procedure call. When the call completes the rectangle is cleared. This is a network activity indicator for the user.

Command Format

```
net blip[on|off]
```

When used without any arguments, the command displays whether blip is on or off.

Examples

```
C>net blip
NFS020I : Blip (on-screen network activity monitor) is enabled.
C>
```

net name — Perform NFS User Authentication

The `net name` command logs the user into the network, and establishes the user's network-wide user id (*uid*) and group id (*gid*). With no arguments, it displays current network and user information.

Command Format

```
net name [ * | username [ * ] ]
```

where:

username, if specified, is your login name. If you enter an *, the command prompts you to supply your username.

You must use * in place of the password. The command prompts you to supply the actual password, which does not display as you enter it.

If you use the supplied configuration program, it asks you for your user name, and puts a `net name` *your-username* * command into the file `\NFS\NETWORK.BAT`, which runs whenever you boot your PC.

When used with no arguments, `net name` displays the following information:

- the name and Internet address of your PC
- the Yellow Pages (YP) domain name and the name and Internet address of any Yellow Pages server
- the name and Internet address of the authentication server.
- the name and Internet address of any gateway system
- your user name and user and group ids
- the date, time, and timezone.

For the `net name` command to work, the `pcnfsd` daemon must be installed and running normally on the currently-selected Yellow Pages server machine. If you have Yellow Pages, PC-NFS looks for the `pcnfsd` server on the Yellow Pages server. If there is no Yellow Pages server machine, the daemon can run on any server. Use the `net ypset` or `net pcnfsd` commands to set the name of the server running the `pcnfsd` daemon.

net pcnet — Run PC-NFS With IBM PC-Network

The `net pcnet` command enables or disables PC-NFS compatibility with IBM PC-Network.

Command Format

```
net pcnet [on|off]
```

Options

When used without any arguments, the command displays whether PC-Network compatibility is enabled or disabled.

Examples

```
C>net pcnet
PC-NET compatibility is OFF.

C>
```

net use — Mount Remote File Systems

The net use command mounts a remote file system on a logical drive, associates a network printer with a DOS printer device, or displays the mounted drives and printer associations.

The net use command corresponds to the UNIX mount facility.

Command Format

net use *drive*: \\hostname\path [/SHARE | /MUSTSHARE | /READONLY]

or

net use *drive*: /d

or

net use *printdevice*: \\hostname\printername

where:

drive: \\hostname\path

mounts a file system on a PC drive letter. In other words, this associates a drive letter on the PC with a directory in the file system on a host (server).

drive is the letter of a PC drive on which to mount the file system. *drive* can be any letter following your last existing disk drive and up to and including the letter S. The drives T, U, and V are reserved for remote printers. Note that it must be followed with a colon (:).

hostname is the name of the host where the file system is located. *path* can be the name of an exported file system, \usr, for instance, or a subdirectory within an exported file system, such as \usr\staff\geoff.

An *exported file system* is a portion of a UNIX file system which the system administrator has made available for other machines to mount.

/MUSTSHARE or /MS mounts the network drive in sharing mode only, to use PC-NFS locking services. This indicates you require Locking Services.

/READONLY or /RO mounts the network drive in read-only mode so that you cannot modify or delete files on the disk.

/SHARE or /SH mounts the network drive in sharing mode, if possible. See Chapter 4 for a discussion of sharing.

drive: `/d`
tells PC-NFS to stop using the disk *drive* that was previously mounted by `net use`. This corresponds to the UNIX `umount` command.

printdevice: \hostname\printername
associates a DOS printer device *printdevice* with a printer on a remote system. *printdevice* must be one of `PRN`,

`LPT1`, `LPT2`, or `LPT3` and must be followed by a colon (`:`). `PRN` is a synonym for `LPT1`. *printername* is the printer name (`lp` , `lw` , and so on) on the server system; in UNIX 4.2, this is the name given in the `/etc/printcap` file on the host *hostname*.

Subsequent `net print` commands or printer output operations to the *printdevice* write spool files into the server's spool directory. (See Chapter 6 for further details.)

With no arguments, the command `net use` displays a list of the currently mounted drives and printers.

Two special names are defined in association with the `net use` command. `$YPSERVER` refers to the current Yellow Pages server system. `$HOME` refers to your home directory (as defined in the `passwd` file on the Yellow Pages server, if you logged in with the `net name` command). Thus in the case where your home directory is on the current (or only) Yellow Pages server, you can use:

```
net use d: \\$YPSERVER\$HOME
```

to mount your home directory on drive *d*.

Examples

To mount the directory `\usr\tjones`, located on the host, `yourhost` on PC drive `E`, enter the following:

```
C>net use e: \\yourhost\usr\tjones
```

To mount the printer on the host `que2` for the PC `percy`, enter the following:

```
C>net use LPT1: \\que2\usr\tmp\percy
```

To see what drives are mounted, enter `net use`. The system displays the file systems and printers that are mounted.

```
C>net use
Drive Filesystem                  Tsize  Kbytes Used  Avail Capacity
D:    \\yourhost\usr\tjones\doc 8192  87544  85120  2424   97%
E:    \\yourhost\usr\tjones     8192  87544  85120  2424   97%
LPT1: \\que2\usr\tmp\percy
```

Notes

Remember to use backslashes instead of slashes when specifying paths to `net use`.

The names $YPSERVER and $HOME must be entirely upper-case.

The drive letters available to you are `A` through `R`. The drive letter `S` is used for certain hardware devices. The drive letters `T`, `U`, and `V` are reserved for remote printers. Drive letters `W`, `X`, `Y` and `Z` are not available because most software packages such as Microsoft WINDOWS cannot use them.

Thegdisk space number shown by `net use` might differ from the number shown by the UNIX `df` command on a host system for the same directory. This difference appears for two reasons. First, the directory might be undergoing change. Second, the UNIX operating system reserves disk space for superuser; this is a configuration parameter that can vary at each site.

To communicate with a VMS server use the alternate syntax:

`net use` ***d*****:** ***host*****:*****path***

in which *path* should not include any slash (/) or backslash (\) characters.

See Also

For more information on file sharing and locking, see Chapter 4.

See the sections in *Installing PC-NFS, A Guide to the User and System Administrator* on the configuration program and mounting drives and printers.

net version — Display PC-NFS Version Number

This command displays the version number of PC-NFS.

Command Format

```
net version
```

Example

```
C>net version
NFS016I: The PC-NFS Version is 3.00
```

rsh Remote Shell

The rsh command executes a command line on a specified host.

Command Format

rsh *host* [-l *username*] *command*

where:

host is the name of the remote server on which to execute the *command.*

command is the command line to execute.

rsh copies its standard input to the remote command, the standard output of the remote command to its standard output, and the standard error of the remote command to its standard error. Interrupt, quit and terminate signals are propagated to the remote command; rsh normally terminates when the remote command does.

Special command line characters that are not quoted are interpreted on the local PC, while quoted special characters are interpreted on the remote machine. Thus the command:

```
C>rsh lizard cat lizard.file > tutorial.file
```

writes the remote file lizard.file from the machine called lizard to the file called tutorial.file on your PC.

The command:

```
C>rsh lizard cat lizard.file ">" another.file
```

writes the file lizard.file on the machine called lizard to the file another.file which also resides on the machine called lizard.

Host names are given in the file \NFS\HOSTS on your PC. Each host has one standard name (the first name given in the file), which is rather long and unambiguous, and optionally one or more nicknames (aliases). If your network uses Yellow Pages, the YP server resolves the *host* name; otherwise you must have an entry for that *host* in your local file \NFS\HOSTS.

Options

-l *username*

Log into the host server using the user name *username.* If not specified, your current network login name is used.

Example – Printing with rsh

You can print files from your PC to a printer on a Sun server by piping the output of a DOS command into a UNIX command on the server. rsh allows you to print files on the machine and printer of your choice. Furthermore, you can use the print command of your choice, such as enscript or lpr. For example:

```
C>type dos_file1 | rsh server_a enscript -2r
C>type dos_file2 | rsh server_a lpr -p
```

Warranties

Sun hereby warrants that it has the right to grant license to use the Licensed Software to Licensee and that it has the right and power to enter into this License.

Sun warrants that the Licensed Software shall substantially conform to its User's Manual, as it exists at the date of delivery, for a period of ninety (90) days from the date of delivery. Sun's sole obligation under this warranty shall be limited to using its best efforts to correct such defects and supply LICENSEE with a corrected version of such Licensed Software as soon as practicable after LICENSEE has notified Sun of such defects. Sun does not warrant that: (1) operation of any of the Licensed Software shall be interpreted or error free, or (2) functions contained in the Licensed Software shall operate in the combinations which may be selected for use by LICENSEE or meet LICENSEE's requirements. Sun's warranty obligations shall be void if the Licensed Software is modified without the written consent of Sun.

Sun warrants any Equipment that may accompany the Licensed Software against defects in workmanship and material, in the country to which Sun ships the Equipment, on a return-to-factory basis for a period of ninety (90) days from the date of delivery. Sun's sole responsibility under this warranty shall be, at Sun's option, to either repair or replace, any component which failed during the warranty period because of a defect in workmanship and material. The stated warranties apply only to the initial end-user, and are contingent upon proper treatment and use of the Equipment and maintenance of a safe and suitable site.

The warranties set forth in this Article are expressly subject to the limitations below.

EXCEPT FOR THE EXPRESSED WARRANTIES STATED HEREIN, SUN DISCLAIMS ALL WARRANTIES ON PRODUCTS FURNISHED HEREUNDER, INCLUDING WITHOUT LIMITATION ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE; and the stated express warranties are in lieu of all obligations or liabilities on the part of Sun arising out of or in connection with the performance of the Licenced Software or Equipment.

LIMITATION OF LIABILITY

SUN DOES NOT MAKE ANY EXPRESSED OR IMPLIED WARANTIES INCLUDING, BUT NOT LIMITED TO, THE WARRANTIES OF DESIGN, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE.

In no event will Sun be liable for any lost revenues or profits of other special, indirect or consequential damages, even if Sun has been advised of the possibility of such damages.

Corporate Headquarters
Sun Microsystems, Inc.
2550 Garcia Avenue
Mountain View, CA 94043
415 960-1300
TLX 287815

For U.S. Sales Office locations, call:
800 821-4643
In CA: 800 821-4642

European Headquarters
Sun Microsystems Europe, Inc.
Sun House
31-41 Pembroke Broadway
Camberley
Surrey GU15 3XD
England
0276 62111
TLX 859017

Germany: (89) 926900-0
UK: 0276 62111
France: (1) 46 30 23 24
Japan: (03) 221-7021

Canadian Headquarters
416 477-6745

Europe, Middle East, and Africa, call European Headquarters:
0276 62111

Elsewhere in the world, call Corporate Headquarters:
415 960-1300
Intercontinental Sales

PC-NFS Commands Quick Reference

Key:

Commands and switches are in `this font`.
Variable arguments are in *this font*.
Optional arguments are enclosed in brackets `[]`.

Note:

Each PC-NFS command takes the `-v` argument, which returns the command's version number. Use `-v` as the only argument.

sun microsystems

Part No. 814-3102-03

PC-NFS Commands Quick Reference

arp — Set and display network addresses.

```
arp   hostname    |  -a    |    -d [ hostname | * ]
    or
arp -s hostname ether_addr   |    -f filename
```

chmod — Change the UNIX protection of an NFS file.

```
chmod [ugoa][+-=][rwxstugo] filename ...
    or
chmod absolute-mode filename ...
```

connect — Connect PC-NFS to network through modem.

```
connect [ -d ] scriptname
```

dos2unix — Convert a file from DOS to UNIX format.

```
dos2unix [ -b | -u | -l ] [-f]  [ input-file [ output-file ] ]
```

ftp — Transfer a file to or from a remote network site.

```
ftp [-b] [-d] [-g] [-i] [ hostname ] [ port ]
```

hangup — Disconnect PC-NFS from moden connection to network.

```
hangup
```

ls — Display information about remote files.

```
ls [-a] [-d] [ -b | -l | -u | -w ] [ filename | directoryname ]...
```

mv — Rename a file

```
mv [drive:]oldname newname
```

myeaddr — This command displays the Ethernet address of your PC.

```
myeaddr controller_type [  ioport_addr | shmem_addr   [ interrupt# ]  ]
```

net blip — Set and display network activity indicator.

```
net blip [ on | off ]
```

net join — Set and display tree structure for remote filesystems.

```
net join [ drive1:\path drive2: ]
    or
net join drive: /d
```

net logout — Reset user id to `nobody`.

```
net logout
```

net name — Set and display information about a PC running PC-NFS.

```
net name [ *  |  username [ password | * ]  ]
```

net pcnet — Run PC-NFS along with IBM PC-Network

```
net pcnet [ on | off ]
```

net pcnfsd — Set and display name of the authentication server.

```
net pcnfsd [ hostname ]
```

net print — Print a file on a network printer.

```
net print [ * | filename... ] [ printdevice ]
```

net route — Set and display name of gateway server.

```
net route [ hostname | /d]
```

net start rdr & net stop rdr — Start up and stop PC-NFS.

```
net start rdr [ my_pc [ * ] ]
    or
net stop rdr
```

net subnet — Set and display subnet mask.

```
net subnet [ mask | * ]
```

net umask — Set the NFS file protection mask.

```
net umask [ octal_value ]
```

net use — Set and display mounted remote filesystems.
Associate a DOS printer with a network printer.

```
net use drive: \\hostname\path [/SHARE] [/MUSTSHARE]
                                     [/READONLY]
    or
net use drive: /d
    or
net use printdevice: \\hostname\printername
```

net version — Display the current version number of PC-NFS.

```
net version
```

net ypdomain — Set and display name of the Yellow Pages domain.

```
net ypdomain [ domainname ]
```

net ypset — Set and display name of Yellow Pages (and authentication)server.

```
net ypset [ hostname | * ]
```

netstat — Display network protocol and interface statistics.

```
netstat [ -s ] [ -i ]
```

nfsping — Display whether a machine is an NFS server.

```
nfsping hostname
```

nfsstat — Display NFS usage statistics.

```
nfsstat [ -c | -n | -r | -b ]
```

rcp — Copy a file between machines on a network.

```
rcp sourcefile destinationfile
    or
rcp [ -r ] sourcefile ...  destination_directory
```

rpcinfo — Display information about remote procedures on other machines.

```
rpcinfo -u hostname program_number version_number
    or
rpcinfo -p hostname
```

rsh — Create remote shell.

```
rsh host [ -l  username ] command
```

showmnt — Display a list of exported filesystems.

```
showmnt [ -e | -l ] hostname
```

unix2dos — Convert a UNIX file to DOS format.

```
unix2dos [ -b | -u | -l ] [ -f ] [ -z ]  [ input_file [ output_file ] ]
```

ypcat — Display values in a Yellow Pages database.

```
ypcat [ -k ] [ -t ] mapname
    or
ypcat -x
```

ypmatch — Find and display an entry in a Yellow Pages database.

```
ypmatch [ -k ] [ -t ] key ... mapname
    or
ypmatch -x
```